



PATENT
Attorney Docket No.: 16869K-086100US
Client Ref. No.: 632/SM

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:

YUTAKA TAKATA et al.

Application No.: 10/632,750

Filed: August 1, 2003

For: DISK CONTROLLER AND
CONTROLLING METHOD OF
THE SAME

Customer No.: 20350

Examiner: Unassigned

Technology Center/Art Unit: Unassigned

Confirmation No.: 4662

**PETITION TO MAKE SPECIAL FOR
NEW APPLICATION UNDER M.P.E.P.
§ 708.02, VIII & 37 C.F.R. § 1.102(d)**

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This is a petition to make special the above-identified application under MPEP § 708.02, VIII & 37 C.F.R. § 1.102(d). The application has not received any examination by an Examiner.

(a) The Commissioner is authorized to charge the petition fee of \$130 under 37 C.F.R. § 1.17(i) and any other fees associated with this paper to Deposit Account 20-1430.

(b) All the claims are believed to be directed to a single invention. If the Office determines that all the claims presented are not obviously directed to a single invention, then Applicants will make an election without traverse as a prerequisite to the grant of special status.

(c) Pre-examination searches were made of U.S. issued patents, including a classification search, a computer database search, and a keyword search. The searches were performed on or around July 12, 2004. The classification search covered Class 710 (subclass 6), Class 711 (subclasses 162 and 170), and Class 714 (subclass 2), and was conducted by a professional search firm, Kramer & Amado, P.C. The computer database search was conducted on the USPTO systems EAST and WEST. The keyword search was conducted in Classes 709 (subclasses 203, 218, 223, and 226), 711 (subclass 112), and 714 (subclasses 5 and 6). The inventors further provided references considered most closely related to the subject matter of the present application (see references #6-11 below), which were cited in the Information Disclosure Statement filed with the application on August 1, 2003.

(d) The following references, copies of which are attached herewith, are deemed most closely related to the subject matter encompassed by the claims:

- (1) U.S. Patent No. 5,768,623;
- (2) U.S. Patent No. 6,449,607 B1;
- (3) U.S. Patent Publication No. 2002/0178336 A1;
- (4) U.S. Patent Publication No. 2003/0105767 A1;
- (5) U.S. Patent Publication No. 2004/0098543 A1;
- (6) Japanese Patent Publication No. 2000-047952;
- (7) Japanese Patent Publication No. 06-332782;
- (8) Japanese Patent Publication No. 2002-163140;
- (9) Japanese Patent Publication No. 2001-051890;
- (10) Japanese Patent Publication No. 2000-207370; and
- (11) Japanese Patent Publication No. 08-335144.

(e) Set forth below is a detailed discussion of references which points out with particularity how the claimed subject matter is distinguishable over the references.

A. Claimed Embodiments of the Present Invention

The claimed embodiments relate to a disk controller and a method of controlling the same to provide high performance.

Independent claim 1 recites a disc controller comprising a network controlling unit configured to receive a data input/output request sent from an external device through a network; and a disc controlling unit formed in the same circuit board in which the network controlling unit is formed, the disc controlling unit coupled to the network controlling unit by an internal bus provided in the circuit board. The disc controlling unit is configured to receive a command sent from the network controlling unit through the internal bus and executes a data input/output for a disc drive in response to the command. The network controlling unit is configured to send the command, for which a plurality of addresses are set, to the disc controlling unit. The disc controlling unit is configured to receive the command and executes data input/output corresponding to each of the addresses set in the command for the disc drive.

Independent claim 13 recites a disc controller comprising a network controlling unit configured to receive a data input/output request sent through a network; and a disc controlling unit formed in the same circuit board in which the network controlling unit is formed, the disc controlling unit being coupled to the network controlling unit by an internal bus provided in the circuit board, receiving a command sent from the network controlling unit through the internal bus, and inputting/outputting data to/from a disc drive in response to the command. The plurality of circuit boards connected so as to be capable of communicating with each other are provided. An occurrence of faults of one of the circuit boards is detected by one of the other circuit boards by exchanging a heartbeat message among the circuit boards. When the occurrence of the faults of one circuit board is detected by one of the other circuit boards, the circuit board different from the circuit board causing the faults substitutes for a processing of the circuit board causing the faults.

Independent claim 14 recites a controlling method of a disc controller having a network controlling unit configured to receive a data input/output request sent from an external device through a network; and a disc controlling unit formed in the same circuit board in which the network controlling unit is formed. The disc controlling unit is connected to the network controlling unit by an internal bus provided in the circuit board, receives a command sent from the network controlling unit through the internal bus, and inputs/outputs data to/from a disc drive in response to the command. The method comprises, by means of the network controlling unit sending one command, for which a plurality of addresses are set, to the disc controlling unit; and by means of the disc controlling unit receiving the command and executing data input/output corresponding to each of the addresses set in this command for the disc drive.

One of the benefits that may be derived is the high speed and high reliability with which the processing of the disk controller can be performed.

B. Discussion of the References

None of the following references disclose or suggest a disc controlling unit formed in the same circuit board in which the network controlling unit is formed, the disc controlling unit coupled to the network controlling unit by an internal bus provided in the circuit board.

1. U.S. Patent No. 5,768,623

This reference discloses an architecture which uses host adapter cards that can reside in the host and can control numerous arrays. A plurality of adapter cards is used. Each adapter has controller functions for a designated storage array. There is a host application interface between an application program running in the host computer 20 and the adapter 22. When a data request is made by an application program to a first adapter A through a host application interface for data that is stored in a storage array not primarily controlled by the first adapter, the data request is communicated through the adapter communication interface

23 to the adapter B primarily controlling the storage array in which the requested data is stored. See column 2, line 45 to column 3, line 30; column 3, line 57 to column 4, line 27.

2. U.S. Patent No. 6,449,607 B1

This reference discloses a disk storage device 100 having a modifiable data management function. The disk storage device is connected to an interface 105 which connects to a network 110. A processor 103 carries out an object management program 350 for converting a control command containing physical address information of the disk storage medium 101 and feeds the converted control command to the disk controller 102. In response to an object management modification request given by the user through the network 110 and the network interface 105, the processor 103 carries out the object management modification program 320 to modify a function of the object management program 350. See column 2, lines 32-65; column 4, lines 18-39.

3. U.S. Patent Publication No. 2002/0178336 A1

This reference discloses a storage subsystem capable of effecting remote copy of write data among a group of storage subsystems without being affected by an increase in the load of data writing by a specific host computer among a plurality of host computers connected to the storage subsystems. The storage subsystem includes a first storage subsystem 1 connected to a plurality of host computers 3 via a first interface 2 and a second storage subsystem 7 connected to the first storage subsystem 1 via a second interface 6 so as to copy write data written in the first storage subsystem from the host computer onto the second storage subsystem from the first storage subsystem, thereby protecting the write data in the first and the second storage subsystems in a multiplex manner. See Figure 1 and [0016]-[0026].

4. U.S. Patent Publication No. 2003/0105767 A1

This reference discloses a method for interfacing of SAN (Storage Area Networks) and NAS (Network Attached Storage), and prevents data miss even when a trouble occurs, and makes it possible that an arbitrary number of NAS interfaces access the same file system with high performance. The storage subsystem 100 includes a plurality of interfaces (110, 120, 130, 140, and 150) for the connection to the external network (600 and 700), a plurality of disks 171 to which the plurality of interfaces are accessible, and a shared memory 180 to which the plurality of interfaces are accessible, wherein the plurality of interfaces are loaded with one of the block interfaces for executing an I/O request in disk blocks, and file interfaces are loaded with file servers for executing an I/O request in files. See Figure 1 and [0016]-[0021].

5. U.S. Patent Publication No. 2004/0098543 A1

This reference relates to a storage subsystem which is capable of performing exclusive control of input/output processing requests without need for imparting to the host processing system. The storage subsystem is comprised of a control unit 12 incorporating a control memory 124, wherein information concerning the extent (range) of an input/output processing request which is transferred from a given one of plural host processors to the control unit upon issuance of the input/output processing request from the former is stored in the control memory with a view to realizing the exclusive control for a plurality of input/output processing requests issues from a plurality of host processors to one logical device. See [0001], [0006]-[0007] and [0024-0027].

6. Japanese Patent Publication No. 2000-047952

This reference discloses a means of efficiently performing I/O processing while minimizing the use of processor, main storage, and system bus resources of a server computer by directly transferring data between a network card and an I/O device such as a network adapter or disk controller. In the network file server system, in processing a remote

file system request by a network card, data is directly transferred between a disk controller and the network card. The number of times the data transfer uses main memory between the disk controller and network card is decreased so that high speed processing is enabled.

7. Japanese Patent Publication No. 06-332782

This reference discloses a technique to prevent the throughput due to the centralization of access requests in a specified file server from plural clients, in a file server system where plural file servers accessing each file storage devices are arranged side by side via a network. The master file server provides a file control means by using a load information table to measure and control the load status of each file server, and a file attribute table that records and controls the file server in charge of access to every file block, selecting a file server where the load is light at the time of writing a file.

8. Japanese Patent Publication No. 2002-163140

This reference discloses a storage system that has a scalability capable of fully coping with the band expansion of a network at a low cost. The storage system is comprised of a storage device capable of storing file data, a plurality of file servers performing file processes in response to requests on file data to the storage device, and a file server managing the transfer processes of the file requests received from clients via an external network to the file servers. An internal network connects the response processes to the clients for the file requests, the storage device, the file servers, and the file server.

9. Japanese Patent Publication No. 2001-051890

This reference discloses a decentralized file server system. The system is equipped with servers decentralized in the network and a virtual decentralized file system mounted on each of the servers. Modules judge whether or not their servers are optimum servers capable of handling requests according to server information holding parts, holding mapping tables between the virtual decentralized file system, all the local file systems, and the server information on all the servers.

10. Japanese Patent Publication No. 2000-207370

This reference discloses a technique to provide a distributed file management system which can make appropriate load distribution by means of plural server computers for generating, referring to, and updating files. The distributed file management system is comprised of server computers, client computer groups, and a network. The server computer contains a storage device which records partial files, a network interface, a partial file management section which controls the write and read of the partial files, a status management section which holds load information, and a distributed file management section.

11. Japanese Patent Publication No. 08-335144

This reference discloses a technique to improve reliability and performance of an external storage device, and to provide non-stop maintenance by distributing a load to the plural storage controllers of redundant configuration. Plural disk drive controllers of redundant configuration for controlling a disk device are connected to a host device by the same SCSI ID. These controllers monitor the mutual operating states and set the load distribution information by interposing a communication mechanism and a common managing table in a normal state. High performance is provided by distributing the load by simultaneously operating the plural disk drive controllers, but in case of fault or maintenance, non-stop operation and non-stop maintenance are provided by executing a switching operation at the degeneracy, and recovery can be achieved by disconnecting on the side of the fault.

(f) In view of this petition, the Examiner is respectfully requested to issue a first Office Action at an early date.

Respectfully submitted,



Chun-Pok Leung
Reg. No. 41,405

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, 8th Floor
San Francisco, California 94111-3834
Tel: 650-326-2400
Fax: 415-576-0300
Attachments
RL:rl
60269491 v1



FEE TRANSMITTAL for FY 2004

Effective 10/01/2003. Patent fees are subject to annual revision.

☐ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT (\$) 130.00

Complete if Known

Application Number	10/632,750
Filing Date	August 1, 2003
First Named Inventor	TAKATA, Yutaka
Examiner Name	Unassigned
Art Unit	Unassigned
Attorney Docket No.	16869K-086100US

METHOD OF PAYMENT (check all that apply)

☐ Check ☐ Credit Card ☐ Money Order ☐ Other ☐ None

☒ Deposit Account:

Deposit Account Number

20-1430

Deposit Account Name

Townsend and Townsend and Crew LLP

The Director is authorized to: (check all that apply)

☒ Charge fee(s) indicated below ☒ Credit any overpayments

☒ Charge any additional fee(s) or any underpayment of fee(s)

☐ Charge fee(s) indicated below, except for the filing fee to the above-identified deposit account.

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Small Entity

Fee Code	Fee (\$)	Fee Code	Fee (\$)	Fee Description	Fee Paid
1001	770	2001	385	Utility filing fee	
1002	340	2002	170	Design filing fee	
1003	530	2003	265	Plant filing fee	
1004	770	2004	385	Reissue filing fee	
1005	160	2005	80	Provisional filing fee	

SUBTOTAL (1)

(\$0.00)

2. EXTRA CLAIM FEES FOR UTILITY AND REISSUE

Total Claims	Extra Claims	Fee from below	Fee Paid
	** =		
Independent Claims	** =		
Multiple Dependent			

Large Entity Small Entity

Fee Code	Fee (\$)	Fee Code	Fee (\$)	Fee Description
1202	18	2202	9	Claims in excess of 20
1201	86	2201	43	Independent claims in excess of 3
1203	290	2203	145	Multiple dependent claim, if not paid
1204	86	2204	43	** Reissue independent claims over original patent
1205	18	2205	9	** Reissue claims in excess of 20 and over original patent

SUBTOTAL (2)

(\$0.00)

**or number previously paid, if greater; For Reissues, see above

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Fee Code	Entity Fee (\$)	Small Fee Code	Entity Fee (\$)	Fee Description	Fee Paid
1051	130	2051	65	Surcharge - late filing fee or oath	
1052	50	2052	25	Surcharge - late provisional filing fee or cover sheet.	
1053	130	1053	130	Non-English specification	
1812	2,520	1812	2,520	For filing a request for reexamination	
1804	920*	1804	920*	Requesting publication of SIR prior to Examiner action	
1805	1,840*	1805	1,840*	Requesting publication of SIR after Examiner action	
1251	110	2251	55	Extension for reply within first month	
1252	420	2252	210	Extension for reply within second month	
1253	950	2253	475	Extension for reply within third month	
1254	1,480	2254	740	Extension for reply within fourth month	
1255	2,010	2255	1,005	Extension for reply within fifth month	
1401	330	2401	165	Notice of Appeal	
1402	330	2402	165	Filing a brief in support of an appeal	
1403	290	2403	145	Request for oral hearing	
1451	1,510	1451	1,510	Petition to institute a public use proceeding	
1452	110	2452	55	Petition to revive - unavoidable	
1453	1,330	2453	665	Petition to revive - unintentional	
1501	1,330	2501	665	Utility issue fee (or reissue)	
1502	480	2502	240	Design issue fee	
1503	640	2503	320	Plant issue fee	
1460	130	1460	130	Petitions to the Commissioner	130
1807	50	1807	50	Petitions related to provisional applications	
1806	180	1806	180	Submission of Information Disclosure Stmt	
8021	40	8021	40	Recording each patent assignment per property (times number of properties)	
1809	770	2809	385	Filing a submission after final rejection (37 CFR § 1.129(a))	
1810	770	2810	385	For each additional invention to be examined (37 CFR § 1.129(b))	
1801	770	2801	385	Request for Continued Examination (RCE)	
1802	900	1802	900	Request for expedited examination of a design application	

Other fee (specify) _____

*Reduced by Basic Filing Fee Paid SUBTOTAL (3)

(\$130.00)

SUBMITTED BY

Complete (if applicable)

Name (Print/Type)	Chun-Pok Leung	Registration No. (Attorney/Agent)	41,405	Telephone	650-326-2400
Signature				Date	August 30, 2004

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-047952

(43) Date of publication of application : 18.02.2000

(51)Int.Cl.

G06F 13/00

G06F 12/00

(21)Application number : 10-211357

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 27.07.1998

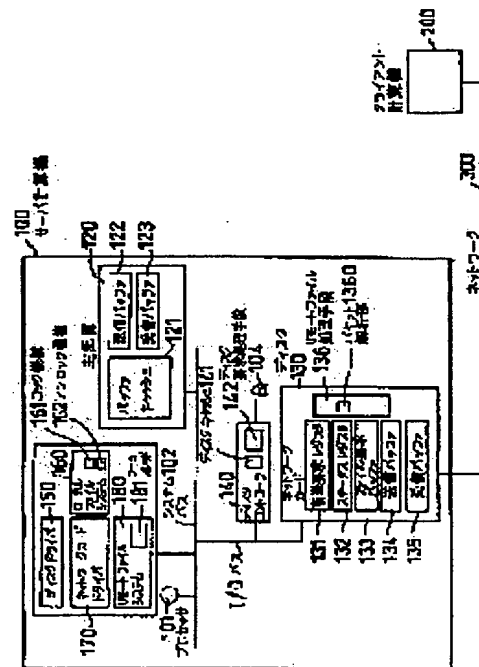
(72)Inventor : TOMOTA MASANORI

(54) NETWORK FILE SERVER SYSTEM AND FILE MANAGING METHOD IN THE SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a means for efficiently performing I/O processing while minimizing the use of processor, main storage and system bus resource of a server computer by directly transferring data between a network card and an I/O device such as network adapter or disk controller.

SOLUTION: In this network file server system, in the case of processing a remote file system request through a remote file processing means 136 by a network card 130 while using a file on a disk through a local file system 160 for an application or operating system(OS) to be operated on a server computer 100, data are directly transferred between a disk controller 140 and the network card 130. Thus, the number of times of data transfer using a main memory 120 between the disk controller 140 and network card 130 is decreased so that high-speed processing is enabled.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C): 1998,2003 Japan Patent Office

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] Manage the above-mentioned storage with the means of communications characterized by providing the following, and the file on this storage is managed. A local file control means with the function to return the information which pinpoints the position of the disk which stores the above-mentioned file to the demand which asks the position on the above-mentioned storage of the above-mentioned file. It has a remote file control means to write the file on the above-mentioned storage based on the content of the remote file system demand from the above-mentioned client computer. The above-mentioned remote file-processing section analyzes the packet which the above-mentioned receive buffer received. The remote file system demand which is an operation demand to the file which the above-mentioned local file control means which the above-mentioned client computer contained in this packet transmitted manages is taken out. When this remote file system demand needs the data transfer from the above-mentioned storage which the above-mentioned local file control means manages, The demand which asks in which position of the above-mentioned storage the file exists is given to the above-mentioned local file control means. A transmitting packet will be created, if it demands to transmit to the above-mentioned storage control means at the above-mentioned transmission buffer based on the positional information which was able to obtain the data of the file concerned and this data is stored in the above-mentioned transmission buffer. The above-mentioned client computer is answered in a remote file system demand. When the above-mentioned remote file system demand needs the data transfer to the above-mentioned storage, The demand which asks in which position of the above-mentioned storage the file concerned exists is given to the above-mentioned local file control means. The data stored in the above-mentioned receive buffer are directly transmitted to the above-mentioned storage control means. The data which the above-mentioned storage control means store in the above-mentioned receive buffer based on the position obtained previously are written in the above-mentioned storage. When the reply to a remote file system demand is performed to the above-mentioned client computer and the above-mentioned remote file system demand does not need the data transfer from the above-mentioned storage The network file server system characterized by passing the demand to the above-mentioned remote file control means. The server computer which offers files, such as various applications, Two or more client computers which access the file which this server computer offers. It is the storage with which it is the network file server system which consists of the network which connects the above-mentioned server computer and the above-mentioned client computer, and the above-mentioned server computer stores information, such as a file. The transmission buffer section which stores the packet which considers protocol processing of the packet transmitted and received to the above-mentioned client computer as the storage control means which control this storage, and transmits to the above-mentioned client computer, the receive buffer section which stores the packet which received from the above-mentioned client computer, and the remote file-processing section which performs processing based on the remote file system demand from the above-mentioned client computer.

[Claim 2] The network file server system which consists of the network which connects the

server computer which is characterized by providing the following, and which offers files, such as various applications, two or more client computers which access the file which this server computer offers, and the above-mentioned server computer and the above-mentioned client computer. The above-mentioned server computer is storage which stores information, such as a file. Storage control means which control this storage. Means of communications which prepared the transmission buffer section which stores the packet which carries out protocol processing of the packet transmitted and received to the above-mentioned client computer, and transmits to the above-mentioned client computer, the receive buffer section which stores the packet which received from the above-mentioned client computer, and the remote file-processing section which performs processing based on the remote file system demand from the above-mentioned client computer. Store the data of the above-mentioned file in the buffer cache section on the primary storage of the above-mentioned server computer, and the operation to the above-mentioned file When it carries out on the above-mentioned buffer cache section and the data of the above-mentioned file are stored on the buffer cache section. The function to return the address of the data of the file concerned on the above-mentioned buffer cache to this file demand. The function to return the information which pinpoints the position of the disk which stores the above-mentioned file to the demand which manages the above-mentioned storage, manages the file on this storage, and asks the position on the above-mentioned storage of the above-mentioned file.

[Claim 3] The network file server system which consists of the network which connects the server computer which is characterized by providing the following, and which offers files, such as various applications, two or more client computers which access the file which this server computer offers, and the above-mentioned server computer and the above-mentioned client computer. The above-mentioned server computer is storage which stores information, such as a file. Storage control means which control this storage. Means of communications which prepared the transmission buffer section which stores the packet which carries out protocol processing of the packet transmitted and received to the above-mentioned client computer, and transmits to the above-mentioned client computer, the receive buffer section which stores the packet which received from the above-mentioned client computer, and the remote file-processing section which performs processing based on the remote file system demand from the above-mentioned client computer. Store the data of the above-mentioned file in the buffer cache section on the primary storage of the above-mentioned server computer, and the operation to the above-mentioned file When it carries out on the above-mentioned buffer cache section and the data of the above-mentioned file are stored on the buffer cache section. The function to return the address of the data of the file concerned on the above-mentioned buffer cache to this file demand. The function to secure the above-mentioned buffer cache section which stores the file concerned based on the above-mentioned file demand, and to return this address when the data of the above-mentioned file are not stored on the buffer cache section. The function to return the information which pinpoints the position of the disk which stores the above-mentioned file to the demand which manages the above-mentioned storage, manages the file on this storage, and asks the position on the above-mentioned storage of the above-mentioned file.

[Claim 4] The network file server system which consists of the network which connects the server computer which is characterized by providing the following, and which offers files, such as various applications, two or more client computers which access the file which this server computer offers, and the above-mentioned server computer and the above-mentioned client computer. The above-mentioned server computer is storage which stores information, such as a file. Storage control means which control this storage. Means of communications which carried out protocol processing of the packet transmitted and received to the above-mentioned client computer, prepared the transmission buffer section which stores the packet which transmits to the above-mentioned client computer, and the receive buffer section which stores the packet which received from the above-mentioned client computer, and prepared the remote file-processing section which processes based on the remote file system demand from the above-mentioned client computer. The function of notifying the positional information on the function of returning the information which pinpoints the position of the disk which stores the above-

mentioned file to the demand which manages the above-mentioned storage, manages the file on this storage, and asks the position on the above-mentioned storage of the above-mentioned file, and the storage store this file in case the writing to the above-mentioned file, deletion, creation, etc. operate, and the file positional information it is the information specify this file to the above-mentioned means of communications.

[Claim 5] Manage the above-mentioned storage with the means of communications characterized by providing the following, and the file on this storage is managed. A local file control means with the function to return the information which pinpoints the position of the disk which stores the above-mentioned file to the demand which asks the position on the above-mentioned storage of the above-mentioned file, it has a remote file control means to write the file on the above-mentioned storage based on the content of the remote file system demand from the above-mentioned client computer. The above-mentioned remote file-processing section analyzes the packet which the above-mentioned receive buffer received. The remote file system demand which is an operation demand to the file which the above-mentioned local file control means which the above-mentioned client computer contained in this packet transmitted manages is taken out. When this remote file system demand needs the data transfer from the above-mentioned storage which the above-mentioned local file control means manages, the demand which asks in which position of the above-mentioned storage the file exists is given to the above-mentioned local file control means. A transmitting packet will be created, if it demands to transmit to the above-mentioned storage control means at the above-mentioned transmission buffer based on the positional information which was able to obtain the data of the file concerned and this data is stored in the above-mentioned transmission buffer. The above-mentioned client computer is answered in a remote file system demand. When the above-mentioned remote file system demand needs the data transfer to the above-mentioned storage, the demand which asks in which position of the above-mentioned storage the file concerned exists is given to the above-mentioned local file control means. The data stored in the above-mentioned receive buffer are directly transmitted to the above-mentioned storage control means. The data which the above-mentioned storage control means store in the above-mentioned receive buffer based on the position obtained previously are written in the above-mentioned storage. When the reply to a remote file system demand is performed to the above-mentioned client computer and the above-mentioned remote file system demand does not need the data transfer from the above-mentioned storage. The file management method in the network file server system characterized by passing the demand to the above-mentioned remote file control means. The server computer which offers files, such as various applications. Two or more client computers which access the file which this server computer offers. It is the storage network which connects the above-mentioned server computer and the above-mentioned client computer, and the above-mentioned server computer stores information, such as a file. The transmission buffer section which stores the packet which considers protocol processing of the packet transmitted and received to the above-mentioned client computer as the storage control means which control this storage, and transmits to the above-mentioned client computer, the receive buffer section which stores the packet which received from the above-mentioned client computer, and the remote file-processing section which performs processing based on the remote file system demand from the above-mentioned client computer.

[Claim 6] The file management method of the network file server system which consists of a network which connects the server computer which is characterized by providing the following, and which offers files, such as various applications, two or more client computers which access the file which this server computer offers, and the above-mentioned server computer and the above-mentioned client computer. The above-mentioned server computer is storage which stores information, such as a file. Storage control means which control this storage. Means of communications which carried out protocol processing of the packet transmitted and received to the above-mentioned client computer, prepared the transmission buffer section which stores the packet which transmits to the above-mentioned client computer, and the receive buffer section which stores the packet which received from the above-mentioned client computer, and prepared the remote file-processing section which processes based on the remote file system demand from the above-mentioned client computer. The function of notifying the positional information on the function of returning the information which pinpoints the position of the disk which stores the above-mentioned file to the demand which manages the above-mentioned storage, manages the file on this storage, and asks the position on the above-mentioned storage of the above-mentioned file, and the storage store this file in case the writing to the above-mentioned file, deletion, creation, etc. operate, and the file positional information it is the information specify this file to the above-mentioned means of communications.

[Translation done.]

the remote file-processing section which performs processing based on the remote file system demand from the above-mentioned client computer. Store the data of the above-mentioned file in the puffer cache section on the primary storage of the above-mentioned server computer, and the operation to the above-mentioned file When it carries out on the above-mentioned buffer cache section and the data of the above-mentioned file are stored on the buffer cache section. The function to return the address of the data of the file concerned on the above-mentioned buffer cache to this file demand. The function to return the information which pinpoints the position of the disk which stores the above-mentioned file to the demand which manages the above-mentioned storage, manages the file on this storage, and asks the position on the above-mentioned storage of the above-mentioned file.

[Claim 7] The file management method of the network file server system which consists of a network which connects the server computer which is characterized by providing the following, and which offers files, such as various applications, two or more client computers which access the file which this server computer offers, and the above-mentioned server computer and the above-mentioned client computer. The above-mentioned server computer is storage which stores information, such as a file. Storage control means which control this storage. Means of communications which prepared the transmission buffer section which stores the packet which carries out protocol processing of the packet transmitted and received to the above-mentioned client computer, and transmits to the above-mentioned client computer, the receive buffer section which stores the packet which received from the above-mentioned client computer, and the remote file-processing section which performs processing based on the remote file system demand from the above-mentioned client computer. Store the data of the above-mentioned file in the puffer cache section on the primary storage of the above-mentioned server computer, and the operation to the above-mentioned file When it carries out on the above-mentioned buffer cache section and the data of the above-mentioned file are stored on the buffer cache section. The function to return the address of the data of the file concerned on the above-mentioned buffer cache to this file demand. The function to secure the above-mentioned file demand, and to return this address when the data of the above-mentioned file are not stored on the buffer cache section. The function to return the information which pinpoints the position of the disk which stores the above-mentioned file to the demand which manages the above-mentioned storage, manages the file on this storage, and asks the position on the above-mentioned storage of the above-mentioned file.

[Claim 8] The file management method of the network file server system which consists of a network which connects the server computer which is characterized by providing the following, and which offers files, such as various applications, two or more client computers which access the file which this server computer offers, and the above-mentioned server computer and the above-mentioned client computer. The above-mentioned server computer is storage which stores information, such as a file. Storage control means which control this storage. Means of communications which carried out protocol processing of the packet transmitted and received to the above-mentioned client computer, prepared the transmission buffer section which stores the packet which transmits to the above-mentioned client computer, and the receive buffer section which stores the packet which received from the above-mentioned client computer, and prepared the remote file-processing section which processes based on the remote file system demand from the above-mentioned client computer. The function of notifying the positional information on the function of returning the information which pinpoints the position of the disk which stores the above-mentioned file to the demand which manages the above-mentioned storage, manages the file on this storage, and asks the position on the above-mentioned storage of the above-mentioned file, and the storage store this file in case the writing to the above-mentioned file, deletion, creation, etc. operate, and the file positional information it is the information specify this file to the above-mentioned means of communications.

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[The technical field to which invention belongs] this invention accesses the file on a server computer through networks, such as a public correspondence network and LAN (Local Area Network) of a cable and radio, from two or more client computers, and relates to the file management method in the network file server system which transmits and receives data, and a network file server system.

[0002]

[Description of the Prior Art] As conventionally shown in drawing 8, in the network file server 10, it was what exchanges information between the client computer 25 and a network file server 10. That is, the packet which stored the file manipulation demand reaches the network card 18 of a network file server 10 (it is hereafter called the server computer 10) via the networks 24, such as LAN, from (1) client computer 25.

[0003] (2) The network driver 20 transmits the packet which reached the network card 18 of the server computer 10 to a primary storage 14 from the receive buffer 19 of the network card 18. [0004] (3) The protocol stack 21 which analyzes communications protocols, such as TCP/IP, analyzes the content of a packet, takes out the file manipulation demand stored in the packet, and passes the remote file system 22.

[0005] (4) The remote file system 22 passes a file manipulation demand to the local file system 23. The local file system 23 is a file system which manages the disk 17 of the server computer 10.

[0006] (5) The local file system 23 processes a file manipulation demand. The result is returned to the remote file system 22.

[0007] (6) The remote file system 22 creates the packet which stored the result, and transmits a result to the client computer 25 via a protocol stack 21, the network card 18, and a network 24. [0008] The software which processes the packet which arrived is the network driver 20, a protocol stack 21, the remote file system 22, and the local file system 23. These shall be installed on storage, such as a disk.

[0009] The processor 11 of the server computer 10 performs such software on a primary storage 14. In processing of a file manipulation demand, the data between a disk 17 and the network card 18 are exchanged via a primary storage 14.

[0010] In the old server computer 10, processing of the protocol stack 21 which realizes reliable data transmission and reception took time, and many processor resources were consumed. [0011] Moreover, in network file service etc., an exchange of data occurs using a disk controller 16 and the network card 18. Both of the devices are connected to I/O bus 13 (it realizes by the PCI bus etc.) of the server computer 10.

[0012] Therefore, after the software (the processor of a server computer performs) which controls the device of the network card 18 and disk controller 16 grade processed by once transmitting data to a primary storage until now, data were further transmitted to each device.

[0013]

[Problem(s) to be Solved by the Invention] If it was in the conventional server computer 10 as

mentioned above, processing of the protocol stack 21 which realizes reliable data transmission and reception took time, and many processor resources were consumed. Moreover, in network file service etc., it was what an exchange of data generates using a disk controller 16 and the network card 18. Both of the devices are connected to I/O bus 13 (it realizes by the PCI bus etc.) of the server computer 10.

[0014] Therefore, after the software (the processor of a server computer performs) which controls the device of the network card 18 and disk controller 16 grade processed by once transmitting data to a primary storage conventionally, data were further transmitted to each device.

[0015] Then, in consideration of the above-mentioned situation, accomplished this invention, and it cancels the above-mentioned fault. The file on a server computer is accessed through networks, such as a public correspondence network and LAN of a cable and radio, from two or more client computers. By being in the network file server system which transmits and receives data, and exchanging data directly among I/O devices, such as a network adapter and a disk controller. Use of the processor of a server computer, a primary storage, and system bus resources is suppressed to the minimum, and it aims at offering the file management method in the network file server system which can perform I/O processing efficiently, and a network file server system.

[0016]

[Means for Solving the Problem] this invention in order to attain the above-mentioned purpose the network file server system of this invention. The server computer which offers files, such as various applications, and two or more client computers which access the file which this server computer offers. It is the network file server system which consists of a network which connects the above-mentioned server computer and the above-mentioned client computer, the above-mentioned server computer. The storage which stores information, such as a file, and the storage control means which control this storage. Protocol processing of the packet transmitted and received to the above-mentioned client computer is carried out. The packet which transmits to the above-mentioned client computer. The means of communications which prepared the transmission buffer section to store, the receive buffer section which stores the packet which received from the above-mentioned client computer, and the remote file-processing section which performs processing based on the remote file system demand from the above-mentioned client computer. A local file control means with the function to return the information which pinpoints the position of the disk which stores the above-mentioned file to the demand which manages the above-mentioned storage, manages the file on this storage, and asks the position on the above-mentioned storage of the above-mentioned file, it has a remote file control means to write the file on the above-mentioned storage based on the content of the remote file system demand from the above-mentioned client computer. The above-mentioned remote file-processing section analyzes the packet which the above-mentioned receive buffer received. The remote file system demand which is an operation demand to the file which the above-mentioned local file control means which the above-mentioned client computer contained in this packet transmitted manages is taken out. When this remote file system demand needs the data transfer from the above-mentioned storage which the above-mentioned local file control means manages, the demand which asks in which position of the above-mentioned storage the file exists is given to the above-mentioned local file control means. A transmitting packet will be created, if it demands to transmit to the above-mentioned storage control means at the above-mentioned transmission buffer based on the positional information which was able to obtain the data of the file concerned and this data is stored in the above-mentioned transmission buffer. The above-mentioned client computer is answered in a remote file system demand. When the above-mentioned remote file system demand needs the data transfer to the above-mentioned storage, the demand which asks in which position of the above-mentioned storage the file concerned exists is given to the above-mentioned local file control means. It demands to transmit to the above-mentioned storage control means at the above-mentioned transmission buffer based on the positional information which was able to obtain the data of a file. If this data is stored in the above-mentioned transmission buffer, will create a transmitting packet and the above-mentioned

client computer will be answered in a remote file system demand. When the above-mentioned remote file system demand needs the data transfer to the above-mentioned storage, the demand which asks in which position of the above-mentioned storage the file concerned exists is given to the above-mentioned local file control means. The data stored in the above-mentioned receive buffer are directly transmitted to the above-mentioned storage control means. The data which the above-mentioned storage control means store in the above-mentioned receive buffer based on the position obtained previously are written in the above-mentioned storage. When the reply to a remote file system demand is performed to the above-mentioned client computer and the above-mentioned remote file system demand does not need the data transfer from the above-mentioned storage, it is characterized by constituting so that the demand may be passed to the above-mentioned remote file control means.

[0017] In case a network card processes a remote file system demand, the application and the operating system (OS) which operate on a server computer using the file on a disk through a local file system according to such composition, transmitting immediate data between a disk controller and a network card can perform high-speed processing by reducing the number of times of the data transfer using the primary storage between a disk controller and a network card.

[0018] Furthermore, only a Network File System demand is sent to the remote file system (the processor of a server computer performs) proposed by this invention among the packets which the network card interpreted. At this time, the number of times of data transfer can be reduced from before by sending only a demand to the primary storage of a server computer, and carrying out the direct DMA transfer of the data of a file etc. between a network card and a disk controller.

[0019] If the above-mentioned local file control means is equipped with the function which notifies the positional information on the storage which stores a file with the demand, and the file positional information which is information which specifies this file to the above-mentioned means of communications in addition to this composition, the disk positional information of a file can be obtained still at high speed.

[0020] Moreover, the application and OS which operate on a server computer using the file on a disk through a local file system in addition to this composition in case a network card processes a remote file system demand, the data of the buffer cache on the primary storage which a local file system manages by transmitting to a network card directly can perform high-speed processing and further an exchange of data with the disk generated in the remote file system demand by storing in a buffer cache Next, the data on a buffer cache can be used to the demand (the application on a server computer lets a local file system pass, or a client computer is as a remote file system demand) to the same file.

[0021]

[Embodiments of the Invention] Hereafter, the gestalt of 1 operation of this invention is explained with reference to a drawing.

[0022] Drawing 1 is the block diagram having shown the network file server structure of a system concerning this operation gestalt. Composition required for explanation of this operation gestalt is shown, and other composition is omitted.

[0023] The networks 300, such as a public correspondence network and LAN of a cable and radio, connect with the server computer 100, and the client computer 200 accesses the file on the server computer 100. This file is a file corresponding to the various applications installed on the server computer, or other applications.

[0024] A processor 101 controls each part of the server computer 100.

[0025] A system bus 102 connects and carries out various data transfer of primary-storage 120 grade and the processor 101.

[0026] I/O bus 103 connects I/O equipment and system buses 102, such as input meanses, such as the network card 130, a disk controller 140, a keyboard that is not illustrated, and a mouse, and a display, and transmits various data.

[0027] A primary storage 120 is a primary storage of the server computer 100, and is equipped with the buffer cache 121, a transmission buffer 122, and a receive buffer 123.

[0028] The remote file system 180, the local file system 160, the network card driver 170, and a disk driver 150 are software, are saved on the storage of disk 104 grade, and are performed by the processor 101.

[0029] The remote file system 180 processes writing the file on the server computer 100 based on the content of a file system demand by the worker thread 181 etc.

[0030] The local file system 160 manages the disk 104 with which the server computer 100 is equipped, and is equipped with the lock function 161 and the unlocking function 162. About the function of the lock function 161 and the unlocking function 162, it mentions later.

[0031] The network card driver 170 controls the network card 130.

[0032] A disk driver 150 controls a disk 104 by the disk controller 140.

[0033] The server computer 100 is equipped with the network card 130 which performs protocol processing for performing transmission and reception of the client computer 200 and data. As a result of performing protocol processing of the packet transmitted and received between the server computer 100 and the client computer 200, when the packet is not the file system demand whose R/W etc. carries out the file on the server computer 100 from the client computer 200, this network card 130 cooperates with the network card driver 170, and performs the same processing as the usual network card.

[0034] Moreover, the network card 130 processes writing the file on the server computer 100 based on the content of a file system demand etc. in harmony with the network card driver 170 which operates on the server computer 100, and the worker thread 181 on the remote file system 180, when the above-mentioned packet which carries out transmission and reception is a file system demand.

[0035] The above-mentioned network card 130 consists of the following.

[0036] Perform protocol processing of a receive packet, and when the receive packet is a file system demand The file demand buffer 133 and receive packet which store the content of the packet which a remote file-processing means 136 to perform the processing, and the remote file-processing means 136 analyzed in the usual packet Or it consists of the status register 132 which stores the information which shows a file system demand etc., the receive buffer 135 which once stores the packet transmitted from the alien machine, a transmission buffer 134 which accumulates the packet which should transmit, and a transfer-request register 131.

[0037] Data transfer of the transfer-request register 131 is carried out to the address of the storing origin of data from there, and the information on the address of the point which stores the data is stored. For example, when transmit data is on the buffer cache 121, the address on the buffer cache 121 of the data and the address on the transmission buffer 134 which is the data transfer point are stored. When received data are on a receive buffer 135, the address on the receive buffer 135 of the data and the address on the buffer cache 121 which is the data transfer point are stored.

[0038] The value which shows the status stored in a status register 132 is carried out as follows here. 1 shows read/write demand reception of a file system, 2 usually shows packet reception, and 3 shows DMA transfer completion, and 4 shows the usual packet transmitting processing. [0039] The file demand buffer 133 consists of the remote file system demand 1330 (a lead, a light, in addition to this), the target file name 1331, its offset 1332, a data length 1333, and a data address 1334, as shown in drawing 2.

[0040] The local file system 160 manages the disk with which the server computer 100 is equipped. That is, a file is stored on a disk 104 and functions, such as creation of a file, a lead, a light, and deletion, are offered. The buffer cache 121 is created on a primary storage 120, and the cache of the data of a file is carried out. Furthermore, it has the following functions.

[0041] 1. Judge whether the content to which the file corresponds is in the buffer cache 121 by considering a file name, offset, and a data length as an input. In being in the buffer cache 121, it returns the address of the buffer.

[0042] 2. Lock function 161 to forbid file manipulation demand to corresponding buffer by considering the address of buffer as input, and unlocking function 162 to cancel it.

[0043] 3. Return the position (disk block) of the disk which stores the corresponding data by considering a file name, offset, and a data length as an input.

[0044] A disk controller 140 is equipped with the disk cache 141 which once stores the disk demand processing means 142 and the data which read from the disk 104 or are written in a disk 104.

[0045] The network card driver 170 which the server computer 100 performs is equipped with the following functions.

[0046] 1. Process interruption from the network card 130.

[0047] 2. Transmit and receive the usual packet.

[0048] 3. Secure a transmission buffer 122 and a receive buffer 123 on a primary storage 120.

[0049] The worker thread 181 which the processor 101 of the server computer 100 performs receives a remote file system demand, and takes charge of the processing.

[0050] Processing begins from the place where, as for a remote file system demand, the client computer 200 transmits a demand to the server computer 100. Hereafter, the client computer 200 advances a remote file system demand, and the flow chart of drawing 3 explains how the server computer 100 processes the demand.

[0051] First, if the remote file-processing means 136 of the network card 130 has reception of a packet from the client computer 200, it will perform the following operation. Here, the example which uses Ethernet for communication and uses TCP/IP for the protocol is explained.

[0052] When the packet addressed to network card 130 is sent on a network 300, the packet is accumulated to a receive buffer 135 (Step A1).

[0053] The packet accumulated to the receive buffer 135 is analyzed, and the packet analysis section 1360 which performs protocol processing of TCP/IP analyzes the data division of a packet, and judges whether it is the protocol which can be processed (Step A2).

[0054] In the case of the protocol which cannot be processed (No of Step A2), Step A8 is processed.

[0055] In the case of the protocol which can be processed (Yes of Step A2), the packet analysis section 1360 performs protocol processing (Step A3).

[0056] Next, it judges whether it is the demand to the remote file system 180 (Step A4). Here, since it is transmitted to the port of TCP/IP defined beforehand, the demand to the remote file system 180 can be easily judged, if the port number of a packet is supervised.

[0057] In the demand to the remote file system 180, (Yes of Step A4) and a status register 132 are set to 1 (read/write demand reception of a file system) (step A5).

[0058] A file system demand is taken out from a packet and it stores in the file demand buffer 133. The demand consists of file manipulation, a file name, offset, a data length, etc. Furthermore, when a demand is a lead, the transmission buffer 134 on the network card 130 is secured, and the address is stored in the data address of the file demand buffer 133. In a light demand, the address of the receive buffer 135 on the network card 130 which stored the data which should be written in is stored in the data address of the file demand buffer 133 (Step A6).

[0059] Then, interruption is applied to a processor (Step A7).

[0060] When it is not the demand to the remote file system 180, in file system demands other than a lead and light processing, a packet is transmitted to the receive buffer 135 on the primary storage 120 which the network card driver 170 secured beforehand as (No of Step A4), and a usual packet. A status register is set to 2 (usually packet reception) (Step A8), and interruption is applied to a processor (Step A7).

[0061] When it is processing with much data transfer, data are directly exchanged between the network card 130 and a disk controller 140 by the method of proposing by this invention. In explanation of this example, although only the lead and the light demand are treated, in addition if there is a demand with the large amount of data transfer, it will process similarly.

[0062] Next, processing operation of the DMA transfer of the network card 130 is explained with reference to the flow chart of drawing 4.

[0063] Data are sent to the transmission place address by the DMA transfer from the transmitting agency address set to the transfer-request register 131 (storing) (Step B1).

[0064] A status register 132 is set to 3. 3 shows DMA transfer completion (step B-2).

[0065] Interruption is applied to a processor (Step B3).

[0066] Next, the flow chart of drawing 5 explains processing operation of transmission of the

network card 130.

[0067] By the packet analysis section 1360, a packet judges whether it is the usual packet transmission (Step C1).

[0068] In the usual packet transmission (Yes of Step C1), the usual packet transmission is performed (Step C2). Then, a status register 132 is set to 4 (Step C3 (4 shows the usual packet transmitting processing)), and interruption is applied to a processor (Step C4).

[0069] When it is not the usual packet transmission (No of Step C1), the data 134 of a transmission buffer and the answer to the client ANTO computer 200 from a file demand are created (Step C5). Protocol processing is performed by the packet analysis section 1360 about this created packet (Step C6). A packet is transmitted to a network (Step C7).

[0070] The network card driver 170 is network card control software which the processor 101 of the server computer 100 performs. The flow chart of drawing 6 explains the processing operation.

[0071] A processor receives interruption and the interrupt handler of the network card driver 170 is called.

[0072] A network card driver interrupt handler judges the demand to the remote file system 180, and the usual packet by investigating the status register of the network card 130 (Step D1).

[0073] When it is not the demand to the remote file system 180 (i.e., when the value stored in the status register 132 is except one), (No of Step D1) and the interrupt handler of the network card driver 170 carry out the same operation as the usual network card driver 170.

[0074] Here, a status register 132 judges whether it is 2 (Step D4).

[0075] When the value stored in the status register 132 is 2 (Yes of Step D4), the interrupt handler of the network card driver 170 removes a header (Ethernet frame information) unnecessary to protocol processing of a high order among the data of the packet which the network card 130 transmitted etc., and passes it to protocol drivers (TCP/IP etc.) (Step D5). Since it becomes the same processing as the usual packet reception henceforth, explanation is omitted here.

[0076] When the value stored in the case 132 of a demand, i.e., status register, to the remote file system 180 is 1, (Yes of Step D1) and the interrupt handler of the network card driver 170 take out a file demand from the file demand buffer 133 of the network card 130 (Step D2).

[0077] The worker thread 181 which processes this remote file system demand is started, and a demand is passed (Step D3).

[0078] When the value stored in the status register 132 is except two (No of Step D4) (however, 1 is already excepted), the value of a status register 132 judges whether it is 3 (Step D6).

[0079] When the value of a status register 132 is 3 (Yes of Step D6), the thread which is demanding the DMA transfer is started (Step D7).

[0080] When the value of a status register 132 is except three (No of Step D6) (however, 1 and 2 are already excepted), the value of a status register 132 judges whether it is 4 (Step D8).

[0081] Processing is ended when the value of a status register 132 is except four (No of Step D8) (however, 1, and 2 and 3 are already excepted).

[0082] When the value of a status register 132 is 4 (Yes of Step D8), the usual completion processing of packet transmitting is performed (Step D9).

[0083] By the above and processing operation of the flow chart of drawing 6, interruption processing of the network card driver 170 is completed now, and the worker thread 181 (remote file system processing is performed) which a processor performs henceforth a series of processing operation explained below.

[0084] This processing operation is explained with reference to the flow chart of drawing 7.

[0085] The passed file manipulation boils the worker thread 181 in a lead, a light, and other operations, and therefore, it performs the following processings.

[0086] First, the file manipulation to which the worker thread 181 was passed judges whether it is a lead demand (Step E1).

[0087] In a lead demand (Yes of Step E1), the worker thread 181 investigates whether a portion to lead exists in the buffer cache 121 on a primary storage 120 from the file of a file demand, offset, and a data length using the function of the local file system 160 (Step E2).

[0088] When a buffer exists (Yes of Step E2), the address of the corresponding buffer can be acquired.

[0089] The following processings are performed when it exists in the buffer cache 121.

[0090] Operation which locks the corresponding buffer is performed using the function of the local file system 160 (Step E3). When the new operation demand to the buffer comes by this operation, it means performing the exclusion delayed in processing of the demand.

[0091] The worker thread 181 sets the address of the data on the buffer cache 121 which wants to lead the transfer-request register 131, and the address on the transmission buffer 134 of the network card 130 which stores this data (storing), gives the demand which carries out the DMA transfer of the content of the buffer of the buffer cache 121 to the network card 130 to the remote file-processing means 136 of the network card 130 based on this stored address, and waits for the completion (Step E4).

[0092] The remote file-processing means of the network card 130 carries out the DMA transfer of the data which the address of the buffer on the primary storage 120 which prepared the buffer for the transmission buffer 134 on the network card 130, and was passed to it points out in response to the above-mentioned DMA demand (Step E5).

[0093] After a DMA transfer is completed, the remote file-processing means 136 of the network card 130 sets a status register 132 to 3 (DMA transfer completion), and applies interruption to a processor (Step E6).

[0094] The interruption routine of the network card driver 170 is called. An interruption routine recognizes that it is interruption of DMA transfer completion by reading a status register 132. The worker thread 181 for which it is waiting by processing of Step E5 is started, and interruption processing is ended (Step E7).

[0095] The worker thread 181 advances a packet Request to Send to the remote file-processing means 136 of the network card 130 (Step E8).

[0096] Since the data to a remote file system demand (lead demand) are contained in the transmission buffer on the network card 130 by the transfer of (2-3), the rest should just perform transmission.

[0097] The remote file-processing means 136 of the network card 130 creates the header of an Ethernet packet, in order to return the data of a transmission buffer 134 to a transmitting agency (client computer 200 which advanced the lead demand), and it performs transmission (Step E9).

[0098] The worker thread 181 cancels the lock to the buffer of the buffer cache 121 of a primary storage 120 using the function of the local file system 160 (Step E10). Henceforth, other threads, a process, etc. can perform operation to the corresponding buffer.

[0099] The following operations are performed when it does not exist in a buffer (No of Step E2).

[0100] The worker thread 181 investigates the disk block which is needed by read-out operation of a file using the function of the local file system 160. It investigates whether the data required of which portion on a disk exist from a file name and the information on offset. Since the position on a disk 104 is usually managed by the disk block number etc., a disk block number is obtained here (Step E11).

[0101] The worker thread 181 reads the disk block obtained at Step E11, and in order to perform a DMA transfer to the transmission buffer 134 on the network card 130 passed from the interruption routine of the network card driver 170, it calls a disk driver 150 (Step E12).

[0102] A disk driver 150 tells the demand to a disk controller 140, and waits for completion (Step E13).

[0103] The disk demand processing means of a disk controller 140 once reads data from a disk 104, and stores the data in a disk cache 141. The cache data is transmitted to the transmission buffer 134 of the network card 130 to which it was directed at Step E12 (Step E14).

[0104] The disk demand processing means 142 of a disk controller 140 applies interruption to a processor 101 (Step E15).

[0105] The interruption routine of a disk driver 150 starts the worker thread 181 which is Step E13 and is waiting for the completion, and ends interruption processing (Step E16).

[0106] The worker thread 181 advances a data Request to Send to the network card 130 (Step E17).

[0107] This is because transmission could be directed on the network card 130 since the data of the remote file system demand (lead demand) demanded from the client computer 200 were ready on the network card 130.

[0108] The data included in the transmission buffer 134 are used for the remote file-processing means 136 of the network card 130, it creates the answer of a remote file system demand, and transmits it to the client computer 200 (Step E18).

[0109] Using the function of the local file system 160, the worker thread 181 issues read-out operation of a file, and waits for the completion (Step E19).

[0110] Originally there is no need of processing this file read-out demand. The lead demand demanded from the client computer 200 is because it has already processed. However, when the data read from the disk are stored in the buffer cache 121 of the server computer 100 and there is the next access, since the cache effect is expectable by enabling it to reuse, you may store data in the buffer cache 121.

[0111] Then, processing of the worker thread 181 is ended.

[0112] By lead demand, when there is nothing (No of Step E1), in a light demand, a light demand judges whether it is a write-through demand (Step E20).

[0113] In a write-through demand (Yes of Step E20) (i.e., the case of the demand which writes data in a disk immediately), the following processings are performed.

[0114] From the file of a file demand, offset, and a data length, the worker thread 181 calls the function of the local file system 160, and investigates whether the portion which should be carried out a light exists in the buffer cache 121 on a primary storage (Step E21).

[0115] Searching for the address of the buffer with which the local file system 160 corresponds when a buffer exists (Yes of Step E21), the worker thread 181 cancels the buffer using the function of the local file system 160 (Step E22). Since data are overwritten by the light demand which will process this from now on, the data on a primary storage 120 are because it is not necessary to return to a disk 104.

[0116] Using the function of the local file system 160, the worker thread 181 investigates a disk block and obtains the disk block which should carry out a light from the file of a file demand, offset, and a data length (Step E23).

[0117] The worker thread 181 advances a demand and waits for the completion so that the data on the receive buffer 135 of the network card 130 may be written in the disk block obtained at Step E23 using a disk driver 150 (Step E24).

[0118] The disk demand processing means 142 of a disk controller 140 applies interruption to a processor 101, after ending the writing of a disk 104 (Step E25).

[0119] The interrupt handler of a disk driver 150 starts the worker thread 181 which is waiting for completion at Step E24, and ends interruption processing (Step E26).

[0120] Since processing completed the worker thread 181, a demand is given for transmitting the answer to a remote file system demand to the client computer 200 to the network card 130 (Step E27).

[0121] The remote file-processing means 136 of the network card 130 transmits completion of a remote file system demand to the client computer 200 (Step E28).

[0122] The worker thread 181 inspects specification of light processing (Step E29).

[0123] Processing will be ended if it is the mode in which light processing does not use the buffer cache 121 (No of Step E29).

[0124] In order to put data into the buffer cache 121 in the case of the mode in which light processing uses a buffer (Yes of Step E29), the function of the local file system 160 is used for the worker thread 181, it processes a lead demand, and waits for the completion (Step E30).

[0125] Originally it is not necessary to also perform this lead demand, when processing a remote file system demand, the buffer cache 121 is used --- bending --- it is because it is usually that which is meaningful in the client computer 200.

[0126] Therefore, there may not be Step E30.

[0127] Then, processing of the worker thread 181 is ended.

[0128] The following processings are performed when a light demand is not a write-through (No of Step E20).

[0129] The worker thread 181 judges whether the portion which should be carried out a light exists in the buffer cache 121 on a primary storage from the file of a file demand, offset, and a data length using the function of the local file system 160 (Step E31).

[0130] When a buffer exists (Yes of Step E31), the address of the corresponding buffer can be acquired. The following processings are performed when a buffer exists.

[0131] The worker thread 181 performs processing which locks the buffer obtained at Step E31 using the function of the local file system 160 (Step E32).

[0132] The worker thread 181 gives a demand to the remote file-processing means 136 of the network card 130, and waits for the completion so that the address of the data in a receive buffer 135 and the address on the buffer cache 121 may be set to the transfer-request register 131 and the DMA transfer of the data in the receive buffer 135 on the network card 130 may be carried out based on this address to the address on the buffer cache 121 obtained at Step E31 (Step E33).

[0133] By this transfer, the data on the buffer cache 121 will be transposed to the data of a remote file demand (light demand).

[0134] If a DMA transfer is completed, the remote file-processing means 136 of the network card 130 will set a status register to 3, and will apply interruption to a processor 101 (Step E34).

[0135] The interruption routine of the network card driver 170 starts the worker thread 181 for which it waits at Step E33, and completes interruption (Step E35).

[0136] A worker thread cancels the lock of the buffer on 181 and the locked buffer cache 121 (Step E36).

[0137] Then, processing of the worker thread 181 is ended.

[0138] The following processings are performed when there is no buffer (No of Step E31).

[0139] Using the function of the local file system 160, the worker thread 181 secures a buffer on the buffer cache 121 (Step E37), and locks the buffer (Step E32).

[0140] After Step E32, the same processing as the case where there is a buffer is continued.

[0141] As mentioned above, conventionally, the data transfer between the network card 130 and a disk once had to accumulate data to the primary storage 120, and had to perform them to it. However, according to this invention, immediate data can be exchanged between the network card 130 and a disk controller 140 because the network card 130, the network card driver 170, a disk driver 150, and a file system process a remote file system demand in cooperation. Moreover, since the exchange of the data is performed maintaining consistency with the buffer of the file system on a primary storage 120, the function in which the local file system 160 on the server computer 100 operates the data on a disk 104 is maintained.

[0142]

[Effect of the Invention] Since according to this invention it is in a network card and a disk controller and remote file manipulation is processed as a full account was given above, performing protocol processing, file processing, etc. for processing remote file manipulation of the processor of a server computer is lost, and it can assign resources and time to the other processing.

[0143] For example, although the same data flowed twice on the system bus with the conventional technology in order in light processing to transmit data to a primary storage from a network card and to transmit data to a disk from a primary storage further, since data are transmitted to a disk only using an I/O bus from a network card, a transfer of the data using a system bus is unnecessary according to this invention

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

- [Drawing 1] The block diagram showing the network file server structure of a system concerning the gestalt of 1 operation of this invention.
- [Drawing 2] Drawing showing the data composition of the file demand buffer concerning the gestalt of this operation.
- [Drawing 3] The flow chart which shows processing operation in the case of reception of the network card of the gestalt of this operation.
- [Drawing 4] The flow chart which shows processing operation in the case of the DMA transfer of the network card concerning the gestalt of this operation.
- [Drawing 5] The flow chart which shows processing operation in the case of transmission of a network card with respect to the gestalt of this operation.
- [Drawing 6] The flow chart which shows processing operation of the interrupt handler of a network card driver with respect to the gestalt of this operation.
- [Drawing 7] The flow chart which shows processing operation of a worker thread with respect to the gestalt of this operation.
- [Drawing 8] Drawing showing the conventional network file server structure of a system.

[Description of Notations]

- 10 --- Server computer
- 11 --- Processor
- 12 --- System bus
- 13 --- PCI bus
- 14 --- Primary storage
- 15 --- Buffer cache
- 16 --- Disk controller
- 17 --- Disk
- 18 --- Network card
- 19 --- Receive buffer
- 20 --- Network driver
- 21 --- Protocol stack
- 22 --- Remote file system
- 23 --- Local file system
- 24 --- Network
- 25 --- Client computer
- 100 --- Server computer
- 101 --- Processor
- 102 --- System bus
- 103 --- I/O bus
- 104 --- Disk
- 120 --- Primary storage
- 121 --- Buffer KYASHU
- 122 --- Transmission buffer

- 123 --- Receive buffer
- 130 --- Network card
- 131 --- Transfer request register
- 132 --- Status register
- 133 --- File demand buffer
- 1330 --- Remote file demand
- 1331 --- File name
- 1332 --- Offset
- 1333 --- Data length
- 1334 --- Data address
- 134 --- Transmission buffer
- 135 --- Receive buffer
- 136 --- Remote file-processing means
- 1360 --- Packet analysis section
- 140 --- Disk controller
- 141 --- Disk KYASHU
- 142 --- Disk demand processing means
- 150 --- Disk driver
- 160 --- Local file system
- 161 --- Lock function
- 162 --- Unlocking function
- 170 --- Network card driver
- 180 --- Remote file system
- 181 --- Worker thread
- 200 --- Client computer
- 300 --- Network
- 1330 --- Remote file demand

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2000-47952
(P2000-47952A)

(43) 公開日 平成12年2月18日 (2000.2.18)

(51) Int.Cl. ⁷	識別記号	F I	テマコード [*] (参考)
G 0 6 F 13/00	3 5 1	G 0 6 F 13/00	3 5 1 E 5 B 0 8 2
12/00	5 1 4	12/00	5 1 4 A 5 B 0 8 9

審査請求 未請求 請求項の数 8 O L (全 18 頁)

(21) 出願番号 特願平10-211357

(22) 出願日 平成10年7月27日 (1998.7.27)

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72) 発明者 友田 正憲

東京都青梅市末広町2丁目9番地 株式会
社東芝青梅工場内

(74) 代理人 100081732

弁理士 大胡 典夫 (外1名)

Fターム (参考) 5B082 FA02 FA12

5B089 AA21 AA22 AC05 AD02 AD06

AED9 AF01 AF02 CA11 CB02

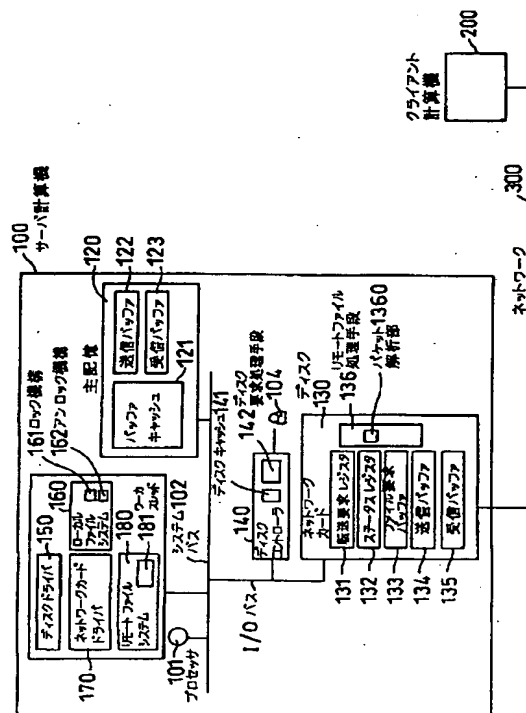
CB03 CB11

(54) 【発明の名称】 ネットワークファイルサーバシステム、及び同システムに於けるファイル管理方法

(57) 【要約】

【課題】 ネットワークアダプタやディスクコントローラ等の I/O デバイス間でデータを直接やりとりすることで、サーバ計算機のプロセッサ、主記憶、システムバス資源の使用を最小限に抑え、効率良く I/O 処理を行う手段を提供すること。

【解決手段】 サーバ計算機 100 上で動作するアプリケーション、OS がディスク上のファイルを、ローカルファイルシステム 160 を通して使用しつつ、ネットワークカード 130 がリモートファイル処理手段 136 により、リモートファイルシステム要求を処理する際、ディスクコントローラ 140 とネットワークカード 130 の間で直接データを転送することで、ディスクコントローラ 140 とネットワークカード 130 間の主記憶 120 を用いたデータ転送の回数を減らすことで、高速な処理を行える。



【特許請求の範囲】

【請求項1】 各種アプリケーション等のファイルを提供するサーバ計算機と、このサーバ計算機の提供するファイルにアクセスする複数のクライアント計算機と、上記サーバ計算機と上記クライアント計算機とを接続するネットワークとから成るネットワークファイルサーバシステムであって、

上記サーバ計算機は、

ファイル等の情報を格納する記憶装置と、

この記憶装置を制御する記憶装置制御手段と、

上記クライアント計算機に送受信するパケットのプロトコル処理をし、上記クライアント計算機へ送信するパケットを格納する送信バッファ部と上記クライアント計算機から受信したパケットを格納する受信バッファ部と上記クライアント計算機からのリモートファイルシステム要求に基づいた処理を行うリモートファイル処理部とを設けた通信手段と、

上記記憶装置を管理しこの記憶装置上のファイルを管理して、上記ファイルの上記記憶装置上の位置を問い合わせる要求に対して上記ファイルを格納しているディスクの位置を特定する情報を返す機能を持つローカルファイル制御手段と、

上記クライアント計算機からのリモートファイルシステム要求の内容に基づき上記記憶装置上のファイルを読み書きするリモートファイル制御手段とを備え、

上記リモートファイル処理部は、

上記受信バッファが受信したパケットを解析し、このパケットに含まれている上記クライアント計算機が送信した上記ローカルファイル制御手段の管理するファイルに対する操作要求であるリモートファイルシステム要求を取り出し、

このリモートファイルシステム要求が上記ローカルファイル制御手段の管理する上記記憶装置からのデータ転送を必要とする場合、上記ローカルファイル制御手段にそのファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記記憶装置制御手段に当該ファイルのデータを得られた位置情報に基づいて上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信パケットを作成して、上記クライアント計算機へリモートファイルシステム要求の返信を行い、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記受信バッファに格納しているデータを上記記憶装置制御手段に直接転送し、上記記憶装置制御手段が先に得た位置に基づき上記受信バッファに格納しているデータを上記記憶装置に書き込み、上記クライアント計算機へリモートファイルシステム要求に対する返信を行い、

上記リモートファイルシステム要求が上記記憶装置からのデータ転送を必要としない場合は、上記リモートファイル制御手段にその要求を渡すことを特徴とするネットワークファイルサーバシステム。

【請求項2】 各種アプリケーション等のファイルを提供するサーバ計算機と、このサーバ計算機の提供するファイルにアクセスする複数のクライアント計算機と、上記サーバ計算機と上記クライアント計算機とを接続するネットワークとから成るネットワークファイルサーバシステムであって、

上記サーバ計算機は、

ファイル等の情報を格納する記憶装置と、

この記憶装置を制御する記憶装置制御手段と、

上記クライアント計算機に送受信するパケットのプロトコル処理をし、上記クライアント計算機へ送信するパケットを格納する送信バッファ部と上記クライアント計算機から受信したパケットを格納する受信バッファ部と上記クライアント計算機からのリモートファイルシステム要求に基づいた処理を行うリモートファイル処理部とを設けた通信手段と、

上記ファイルのデータを上記サーバ計算機の主記憶上のバッファキャッシュ部に格納しておき、上記ファイルに対する操作は、上記バッファキャッシュ部上で行い、上記ファイルのデータをバッファキャッシュ部上に格納している場合には、このファイル要求に対して、上記バッファキャッシュ上の当該ファイルのデータのアドレスを返す機能と、上記記憶装置を管理しこの記憶装置上のファイルを管理して、上記ファイルの上記記憶装置上の位置を問い合わせる要求に対して上記ファイルを格納しているディスクの位置を特定する情報を返す機能とを設けたローカルファイル制御手段と、

上記クライアント計算機からのリモートファイルシステム要求の内容に基づき上記記憶装置上のファイルを読み書きするリモートファイル制御手段とを備え、

上記リモートファイル処理部は、

上記受信バッファが受信したパケットを解析し、このパケットに含まれている上記クライアント計算機が送信した上記ローカルファイル制御手段の管理するファイルに対する操作要求であるリモートファイルシステム要求を取り出し、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記バッファキャッシュ部のどの位置に存在するかを問い合わせる要求をし、

上記バッファキャッシュ部上に存在する場合は、当該ファイルのデータのアドレスに基づいてデータをバッファキャッシュ部から上記送信バッファ部にコピーし、当該データを含む送信パケットを作成して、上記クライアント計算機へリモートファイルシステム要求の返信を行い、

上記バッファキャッシュ部に存在しない場合は、上記ローカルファイル制御手段にそのファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記記憶装置制御手段に当該ファイルのデータを得られた位置情報に基づいて上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信パケットを作成して、上記クライアント計算機ヘリモートファイルシステム要求の返信を行い、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記バッファキャッシュ部のどの位置に存在するかを問い合わせる要求をし、

上記バッファキャッシュ部に存在する場合は、当該ファイルのデータのアドレスに基づいて上記受信バッファのデータを書き込み、上記クライアント計算機ヘリモートファイルシステム要求に対する返信を行い、

上記バッファキャッシュ部に存在しない場合は、上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記受信バッファに格納しているデータを上記記憶装置制御手段に直接転送し、上記記憶装置制御手段が先に得た位置に基づき上記受信バッファに格納しているデータを上記記憶装置に書き込み、上記クライアント計算機ヘリモートファイルシステム要求に対する返信を行い、

リモートファイルシステム要求が上記記憶装置からのデータ転送を必要としない場合は、上記リモートファイル制御手段にその要求を渡すことを特徴とするネットワークファイルサーバシステム。

【請求項3】 各種アプリケーション等のファイルを提供するサーバ計算機と、このサーバ計算機の提供するファイルにアクセスする複数のクライアント計算機と、上記サーバ計算機と上記クライアント計算機とを接続するネットワークとから成るネットワークファイルサーバシステムであって、

上記サーバ計算機は、

ファイル等の情報を格納する記憶装置と、

この記憶装置を制御する記憶装置制御手段と、

上記クライアント計算機に送受信するパケットのプロトコル処理をし、上記クライアント計算機へ送信するパケットを格納する送信バッファ部と上記クライアント計算機から受信したパケットを格納する受信バッファ部と上記クライアント計算機からのリモートファイルシステム要求に基づいた処理を行うリモートファイル処理部とを設けた通信手段と、

上記ファイルのデータを上記サーバ計算機の主記憶上のバッファキャッシュ部に格納しておき、上記ファイルに対する操作は、上記バッファキャッシュ部上で行い、上記ファイルのデータをバッファキャッシュ部上に格納し

ている場合には、このファイル要求に対して、上記バッファキャッシュ上の当該ファイルのデータのアドレスを返す機能と、上記ファイルのデータをバッファキャッシュ部に格納していない場合、上記ファイル要求に基づいて、当該ファイルを格納する上記バッファキャッシュ部を確保し、このアドレスを返す機能と、上記記憶装置を管理しこの記憶装置上のファイルを管理して、上記ファイルの上記記憶装置上の位置を問い合わせる要求に対して上記ファイルを格納しているディスクの位置を特定する情報を返す機能とを設けたローカルファイル制御手段と、

上記クライアント計算機からのリモートファイルシステム要求の内容に基づき上記記憶装置上のファイルを読み書きするリモートファイル制御手段とを備え、上記リモートファイル処理部は、

上記受信バッファが受信したパケットを解析し、このパケットに含まれている上記クライアント計算機が送信した上記ローカルファイル制御手段の管理するファイルに対する操作要求であるリモートファイルシステム要求を取り出し、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記バッファキャッシュ部のどの位置に存在するかを問い合わせる要求をし、

上記バッファキャッシュ部に存在する場合は、当該ファイルのデータのアドレスに基づいてデータをバッファキャッシュ部から上記送信バッファ部にコピーし、当該データを含む送信パケットを作成して、上記クライアント計算機ヘリモートファイルシステム要求の返信を行い、

上記バッファキャッシュ部に存在しない場合は、上記ローカルファイル制御手段にそのファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記記憶装置制御手段に当該ファイルのデータを得られた位置情報に基づいて上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信パケットを作成して、上記クライアント計算機ヘリモートファイルシステム要求の返信を行い、上記ローカルファイル制御手段が確保したバッファキャッシュ部のアドレスに上記データを転送し、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記バッファキャッシュ部のどの位置に存在するかを問い合わせる要求をし、

上記バッファキャッシュ部に存在する場合は、当該ファイルのデータのアドレスに基づいて上記受信バッファのデータを書き込み、上記クライアント計算機ヘリモートファイルシステム要求に対する返信を行い、

上記バッファキャッシュ部に存在しない場合は、上記リモートファイルシステム要求が上記記憶装置へのデー

タ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記受信バッファに格納しているデータを上記記憶装置制御手段に直接転送し、上記記憶装置制御手段が先に得た位置に基づき上記受信バッファに格納しているデータを上記記憶装置に書き込み、上記クライアント計算機ヘリモートファイルシステム要求に対する返信を行い、上記ローカルファイル制御手段が確保したバッファキャッシュ部のアドレスに上記受信バッファに格納しているデータを転送し、リモートファイルシステム要求が上記記憶装置からのデータ転送を必要としない場合は、上記リモートファイル制御手段にその要求を渡すことを特徴とするネットワークファイルサーバシステム。

【請求項4】 各種アプリケーション等のファイルを提供するサーバ計算機と、このサーバ計算機の提供するファイルにアクセスする複数のクライアント計算機と、上記サーバ計算機と上記クライアント計算機とを接続するネットワークとから成るネットワークファイルサーバシステムであって、

上記サーバ計算機は、
ファイル等の情報を格納する記憶装置と、
この記憶装置を制御する記憶装置制御手段と、
上記クライアント計算機に送受信するパケットのプロトコル処理をし、上記クライアント計算機へ送信するパケットを格納する送信バッファ部と上記クライアント計算機から受信したパケットを格納する受信バッファ部とを設け、上記クライアント計算機からのリモートファイルシステム要求に基づき処理を行うリモートファイル処理部とを設けた通信手段と、

上記記憶装置を管理しこの記憶装置上のファイルを管理して、上記ファイルの上記記憶装置上の位置を問い合わせる要求に対して上記ファイルを格納しているディスクの位置を特定する情報を返す機能と、上記ファイルへの書き込み、削除、作成等の操作を行う際に、このファイルを格納する記憶装置上の位置情報とこのファイルを特定する情報であるファイル位置情報を上記通信手段に通知する機能とを備えたローカルファイル制御手段と、
上記クライアント計算機からのリモートファイルシステム要求の内容に基づき上記記憶装置上のファイルを読み書きするリモートファイル制御手段とを備え、上記リモートファイル処理部は、

上記受信バッファが受信したパケットを解析し、このパケットに含まれている上記クライアント計算機が送信した上記ローカルファイル制御手段の管理するファイルに対する操作要求であるリモートファイルシステム要求を取り出し、

このリモートファイルシステム要求が上記ローカルファイル制御手段の管理する上記記憶装置からのデータ転送を必要とする場合、上記ファイル位置情報に基づき一致

するファイルがあるかを検索し、一致した場合、上記記憶装置制御手段に上記ファイル位置情報に基づき上記記憶装置上のデータを上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信パケットを作成して、上記クライアント計算機ヘリモートファイルシステム要求の返信を行い、

一致しない場合、上記ローカルファイル制御手段にそのファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記記憶装置制御手段に当該ファイルのデータを得られた位置情報に基づいて上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信パケットを作成して、上記クライアント計算機ヘリモートファイルシステム要求の返信を行い、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ファイル位置情報に基づき一致するファイルがあるかを検索し、一致した場合、上記ファイル位置情報に基づき上記受信バッファに格納しているデータを上記記憶装置制御手段に転送し、上記記憶装置制御手段は上記ファイル位置情報に基づき上記受信バッファに格納しているデータを上記記憶装置上に書き込み、上記クライアント計算機ヘリモートファイルシステム要求に対する返信を行い、

一致しない場合、上記ローカルファイル制御手段に当該ファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記受信バッファに格納しているデータを上記記憶装置制御手段に直接転送し、上記記憶装置制御手段が先に得た位置に基づき上記受信バッファに格納しているデータを書き込み、上記クライアント計算機ヘリモートファイルシステム要求の返信を行い、
リモートファイルシステム要求が上記記憶装置からのデータ転送を必要としない場合は、上記リモートファイル制御手段にその要求を渡すことを特徴とするネットワークファイルサーバシステム。

【請求項5】 各種アプリケーション等のファイルを提供するサーバ計算機と、このサーバ計算機の提供するファイルにアクセスする複数のクライアント計算機と、上記サーバ計算機と上記クライアント計算機とを接続するネットワークとから成るネットワークファイルサーバシステムに於けるファイル管理方法であって、

上記サーバ計算機は、
ファイル等の情報を格納する記憶装置と、
この記憶装置を制御する記憶装置制御手段と、
上記クライアント計算機に送受信するパケットのプロトコル処理をし、上記クライアント計算機へ送信するパケットを格納する送信バッファ部と上記クライアント計算機から受信したパケットを格納する受信バッファ部と上記クライアント計算機からのリモートファイルシステム要求に基づいた処理を行うリモートファイル処理部とを設けた通信手段と、

上記記憶装置を管理しこの記憶装置上のファイルを管理して、上記ファイルの上記記憶装置上の位置を問い合わせる要求に対して上記ファイルを格納しているディスクの位置を特定する情報を返す機能を持つローカルファイル制御手段と、

上記クライアント計算機からのリモートファイルシステム要求の内容に基づき上記記憶装置上のファイルを読み書きするリモートファイル制御手段とを備え、

上記リモートファイル処理部は、

上記受信バッファが受信したバケットを解析し、このバケットに含まれている上記クライアント計算機が送信した上記ローカルファイル制御手段の管理するファイルに対する操作要求であるリモートファイルシステム要求を取り出し、

このリモートファイルシステム要求が上記ローカルファイル制御手段の管理する上記記憶装置からのデータ転送を必要とする場合、上記ローカルファイル制御手段にそのファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記記憶装置制御手段に当該ファイルのデータを得られた位置情報に基づいて上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信バケットを作成して、上記クライアント計算機へリモートファイルシステム要求の返信を行い、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記受信バッファに格納しているデータを上記記憶装置制御手段に直接転送し、上記記憶装置制御手段が先に得た位置に基づき上記受信バッファに格納しているデータを上記記憶装置に書き込み、上記クライアント計算機へリモートファイルシステム要求に対する返信を行い、

上記リモートファイルシステム要求が上記記憶装置からのデータ転送を必要としない場合は、上記リモートファイル制御手段にその要求を渡すことを特徴とするネットワークファイルサーバシステムに於けるファイル管理方法。

【請求項6】 各種アプリケーション等のファイルを提供するサーバ計算機と、このサーバ計算機の提供するファイルにアクセスする複数のクライアント計算機と、上記サーバ計算機と上記クライアント計算機とを接続するネットワークとからなるネットワークファイルサーバシステムのファイル管理方法であって、

上記サーバ計算機は、

ファイル等の情報を格納する記憶装置と、

この記憶装置を制御する記憶装置制御手段と、

上記クライアント計算機に送受信するバケットのプロトコル処理をし、上記クライアント計算機へ送信するバケットを格納する送信バッファ部と上記クライアント計算

機から受信したバケットを格納する受信バッファ部と上記クライアント計算機からのリモートファイルシステム要求に基づいた処理を行うリモートファイル処理部とを設けた通信手段と、

上記ファイルのデータを上記サーバ計算機の主記憶上のバッファキャッシュ部に格納しておき、上記ファイルに対する操作は、上記バッファキャッシュ部上で行い、上記ファイルのデータをバッファキャッシュ部上に格納している場合には、このファイル要求に対して、上記バッファキャッシュ部上の当該ファイルのデータのアドレスを返す機能と、上記記憶装置を管理しこの記憶装置上のファイルを管理して、上記ファイルの上記記憶装置上の位置を問い合わせる要求に対して上記ファイルを格納しているディスクの位置を特定する情報を返す機能とを設けたローカルファイル制御手段と、

上記クライアント計算機からのリモートファイルシステム要求の内容に基づき上記記憶装置上のファイルを読み書きするリモートファイル制御手段とを備え、上記リモートファイル処理部は、

上記受信バッファが受信したバケットを解析し、このバケットに含まれている上記クライアント計算機が送信した上記ローカルファイル制御手段の管理するファイルに対する操作要求であるリモートファイルシステム要求を取り出し、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記バッファキャッシュ部のどの位置に存在するかを問い合わせる要求をし、

上記バッファキャッシュ部上に存在する場合は、当該ファイルのデータのアドレスに基づいてデータをバッファキャッシュ部から上記送信バッファ部にコピーし、当該データを含む送信バケットを作成して、上記クライアント計算機へリモートファイルシステム要求の返信を行い、

上記バッファキャッシュ部に存在しない場合は、上記ローカルファイル制御手段にそのファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記記憶装置制御手段に当該ファイルのデータを得られた位置情報に基づいて上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信バケットを作成して、上記クライアント計算機へリモートファイルシステム要求の返信を行い、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記バッファキャッシュ部のどの位置に存在するかを問い合わせる要求をし、

上記バッファキャッシュ部上に存在する場合は、当該ファイルのデータのアドレスに基づいて上記受信バッファのデータを書き込み、上記クライアント計算機へリモートファイルシステム要求に対する返信を行い、

上記バッファキャッシュ部に存在しない場合は、上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記受信バッファに格納しているデータを上記記憶装置制御手段に直接転送し、上記記憶装置制御手段が先に得た位置に基づき上記受信バッファに格納しているデータを上記記憶装置に書き込み、上記クライアント計算機へリモートファイルシステム要求に対する返信を行い、

リモートファイルシステム要求が上記記憶装置からのデータ転送を必要としない場合は、上記リモートファイル制御手段にその要求を渡すことを特徴とするネットワークファイルサーバシステムに於けるファイル管理方法。

【請求項7】 各種アプリケーション等のファイルを提供するサーバ計算機と、このサーバ計算機の提供するファイルにアクセスする複数のクライアント計算機と、上記サーバ計算機と上記クライアント計算機とを接続するネットワークとからなるネットワークファイルサーバシステムのファイル管理方法であって、

上記サーバ計算機は、

ファイル等の情報を格納する記憶装置と、

この記憶装置を制御する記憶装置制御手段と、

上記クライアント計算機に送受信するパケットのプロトコル処理をし、上記クライアント計算機へ送信するパケットを格納する送信バッファ部と上記クライアント計算機から受信したパケットを格納する受信バッファ部と上記クライアント計算機からのリモートファイルシステム要求に基づいた処理を行うリモートファイル処理部とを設けた通信手段と、

上記ファイルのデータを上記サーバ計算機の主記憶上のバッファキャッシュ部に格納しておき、上記ファイルに対する操作は、上記バッファキャッシュ部上で行い、上記ファイルのデータをバッファキャッシュ部上に格納している場合には、このファイル要求に対して、上記バッファキャッシュ上の当該ファイルのデータのアドレスを返す機能と、上記ファイルのデータをバッファキャッシュ部上に格納していない場合、上記ファイル要求に基づいて、当該ファイルを格納する上記バッファキャッシュ部を確保し、このアドレスを返す機能と、上記記憶装置を管理しこの記憶装置上のファイルを管理して、上記ファイルの上記記憶装置上の位置を問い合わせる要求に対して上記ファイルを格納しているディスクの位置を特定する情報を返す機能とを設けたローカルファイル制御手段と、

上記クライアント計算機からのリモートファイルシステム要求の内容に基づき上記記憶装置上のファイルを読み書きするリモートファイル制御手段とを備え、

上記リモートファイル処理部は、

上記受信バッファが受信したパケットを解析し、このパ

ケットに含まれている上記クライアント計算機が送信した上記ローカルファイル制御手段の管理するファイルに対する操作要求であるリモートファイルシステム要求を取り出し、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記バッファキャッシュ部のどの位置に存在するかを問い合わせる要求をし、

上記バッファキャッシュ部に存在する場合は、当該ファイルのデータのアドレスに基づいてデータをバッファキャッシュ部から上記送信バッファ部にコピーし、当該データを含む送信パケットを作成して、上記クライアント計算機へリモートファイルシステム要求の返信を行い、

上記バッファキャッシュ部に存在しない場合は、上記ローカルファイル制御手段にそのファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記記憶装置制御手段に当該ファイルのデータを得られた位置情報に基づいて上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信パケットを作成して、上記クライアント計算機へリモートファイルシステム要求の返信を行い、上記ローカルファイル制御手段が確保したバッファキャッシュ部のアドレスに上記データを転送し、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記バッファキャッシュ部のどの位置に存在するかを問い合わせる要求をし、

上記バッファキャッシュ部に存在する場合は、当該ファイルのデータのアドレスに基づいて上記受信バッファのデータを書き込み、上記クライアント計算機へリモートファイルシステム要求に対する返信を行い、

上記バッファキャッシュ部に存在しない場合は、上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記受信バッファに格納しているデータを上記記憶装置制御手段に直接転送し、上記記憶装置制御手段が先に得た位置に基づき上記受信バッファに格納しているデータを上記記憶装置に書き込み、上記クライアント計算機へリモートファイルシステム要求に対する返信を行い、上記ローカルファイル制御手段が確保したバッファキャッシュ部のアドレスに上記受信バッファに格納しているデータを転送し、

リモートファイルシステム要求が上記記憶装置からのデータ転送を必要としない場合は、上記リモートファイル制御手段にその要求を渡すことを特徴とするネットワークファイルサーバシステムに於けるファイル管理方法。

【請求項8】 各種アプリケーション等のファイルを提供するサーバ計算機と、このサーバ計算機の提供するフ

ファイルにアクセスする複数のクライアント計算機と、上記サーバ計算機と上記クライアント計算機とを接続するネットワークとからなるネットワークファイルサーバシステムのファイル管理方法であって、

上記サーバ計算機は、

ファイル等の情報を格納する記憶装置と、

この記憶装置を制御する記憶装置制御手段と、

上記クライアント計算機に送受信するパケットのプロトコル処理をし、上記クライアント計算機へ送信するパケットを格納する送信バッファ部と上記クライアント計算機から受信したパケットを格納する受信バッファ部とを設け、上記クライアント計算機からのリモートファイルシステム要求に基づき処理を行うリモートファイル処理部とを設けた通信手段と、

上記記憶装置を管理しこの記憶装置上のファイルを管理して、上記ファイルの上記記憶装置上の位置を問い合わせる要求に対して上記ファイルを格納しているディスクの位置を特定する情報を返す機能と、上記ファイルへの書き込み、削除、作成等の操作を行う際に、このファイルを格納する記憶装置上の位置情報とこのファイルを特定する情報であるファイル位置情報を上記通信手段に通知する機能とを備えたローカルファイル制御手段と、上記クライアント計算機からのリモートファイルシステム要求の内容に基づき上記記憶装置上のファイルを読み書きするリモートファイル制御手段とを備え、上記リモートファイル処理部は、

上記受信バッファが受信したパケットを解析し、このパケットに含まれている上記クライアント計算機が送信した上記ローカルファイル制御手段の管理するファイルに対する操作要求であるリモートファイルシステム要求を取り出し、

このリモートファイルシステム要求が上記ローカルファイル制御手段の管理する上記記憶装置からのデータ転送を必要とする場合、上記ファイル位置情報に基づき一致するファイルがあるかを検索し、一致した場合、上記記憶装置制御手段に上記ファイル位置情報に基づき上記記憶装置上のデータを上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信パケットを作成して、上記クライアント計算機へリモートファイルシステム要求の返信を行い、

一致しない場合、上記ローカルファイル制御手段にそのファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記記憶装置制御手段に当該ファイルのデータを得られた位置情報に基づいて上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信パケットを作成して、上記クライアント計算機へリモートファイルシステム要求の返信を行い、

上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ファイル位置情報に

基づき一致するファイルがあるかを検索し、一致した場合、上記ファイル位置情報に基づき上記受信バッファに格納しているデータを上記記憶装置制御手段に転送し、上記記憶装置制御手段は上記ファイル位置情報に基づき上記受信バッファに格納しているデータを上記記憶装置上に書き込み、上記クライアント計算機へリモートファイルシステム要求に対する返信を行い、

一致しない場合、上記ローカルファイル制御手段に当該ファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記受信バッファに格納しているデータを上記記憶装置制御手段に直接転送し、上記記憶装置制御手段が先に得た位置に基づき上記受信バッファに格納しているデータを書き込み、上記クライアント計算機へリモートファイルシステム要求の返信を行い、リモートファイルシステム要求が上記記憶装置からのデータ転送を必要としない場合は、上記リモートファイル制御手段にその要求を渡すことを特徴とするネットワークファイルサーバシステムに於けるファイル管理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、複数のクライアント計算機から公衆通信網や有線、無線のLAN (Local Area Network) 等のネットワークを介してサーバ計算機上のファイルにアクセスし、データの送受信を行うネットワークファイルサーバシステム、及びネットワークファイルサーバシステムに於けるファイル管理方法に関する。

【0002】

【従来の技術】従来、図8に示すように、ネットワークファイルサーバ10では、クライアント計算機25とネットワークファイルサーバ10の間で情報をやり取りするものであった。即ち、

(1) クライアント計算機25から、LAN等のネットワーク24を経由し、ファイル操作要求を格納したパケットがネットワークファイルサーバ10 (以下、サーバ計算機10と呼ぶ) のネットワークカード18に届く。

【0003】(2) ネットワークドライバ20が、サーバ計算機10のネットワークカード18に届いたパケットをネットワークカード18の受信バッファ19から、主記憶14に転送する。

【0004】(3) TCP/IP等の通信プロトコルを解析するプロトコルスタック21が、パケットの内容を解析し、パケットに格納されたファイル操作要求を取り出して、リモートファイルシステム22に渡す。

【0005】(4) リモートファイルシステム22は、ファイル操作要求をローカルファイルシステム23に渡す。ローカルファイルシステム23は、サーバ計算機10のディスク17を管理するファイルシステムである。

【0006】(5) ローカルファイルシステム23がファイル操作要求を処理する。その結果をリモートファイ

ルシステム 22 に返す。

【0007】(6) リモートファイルシステム 22 は、結果を格納したバケットを作成し、プロトコスタック 21、ネットワークカード 18、ネットワーク 24 を経由して、結果をクライアント計算機 25 に送信する。

【0008】到着したバケットを処理するソフトウェアは、ネットワークドライバ 20、プロトコスタック 21、リモートファイルシステム 22、ローカルファイルシステム 23 である。これらは、ディスク等の記憶装置上にインストールされているものとする。

【0009】サーバ計算機 10 のプロセッサ 11 が、主記憶 14 上でこれらのソフトウェアを実行する。ファイル操作要求の処理では、主記憶 14 を経由して、ディスク 17 とネットワークカード 18 間のデータのやり取りを行う。

【0010】これまでのサーバ計算機 10 に於いて、信頼性の高いデータ送受信を実現するプロトコスタック 21 の処理に時間がかかり、プロセッサ資源を多く消費していた。

【0011】また、ネットワークファイルサービス等に於いては、ディスクコントローラ 16、ネットワークカード 18 を用いてデータのやりとりが発生する。どちらのデバイスも、サーバ計算機 10 の I/O バス 13 (PCI バス等で実現する) に接続されている。

【0012】そのために、これまでは、ネットワークカード 18、ディスクコントローラ 16 等のデバイスを制御するソフトウェア (サーバ計算機のプロセッサが実行する) が、データを主記憶にいったん転送し、処理を行った後、各デバイスへデータをさらに転送していたものであった。

【0013】

【発明が解決しようとする課題】 上述したように、従来のサーバ計算機 10 にあっては、信頼性の高いデータ送受信を実現するプロトコスタック 21 の処理に時間がかかり、プロセッサ資源を多く消費していた。また、ネットワークファイルサービス等に於いては、ディスクコントローラ 16、ネットワークカード 18 を用いてデータのやりとりが発生するものであった。どちらのデバイスも、サーバ計算機 10 の I/O バス 13 (PCI バス等で実現する) に接続されている。

【0014】そのために、従来は、ネットワークカード 18、ディスクコントローラ 16 等のデバイスを制御するソフトウェア (サーバ計算機のプロセッサが実行する) が、データを主記憶にいったん転送し、処理を行った後、各デバイスへデータを更に転送していたものであった。

【0015】そこで、本発明は上記事情を考慮して成されたもので、上記不具合を解消し、複数のクライアント計算機から公衆通信網や有線、無線の LAN 等のネットワークを介してサーバ計算機上のファイルにアクセス

し、データの送受信を行うネットワークファイルサーバシステムにあって、ネットワークアダプタやディスクコントローラ等の I/O デバイス間でデータを直接やりとりすることで、サーバ計算機のプロセッサ、主記憶、システムバス資源の使用を最小限に抑え、効率良く I/O 処理を行えるネットワークファイルサーバシステム、及びネットワークファイルサーバシステムに於けるファイル管理方法を提供することを目的とする。

【0016】

【課題を解決するための手段】 本発明は上記目的を達成する為、本発明のネットワークファイルサーバシステムは、各種アプリケーション等のファイルを提供するサーバ計算機と、このサーバ計算機の提供するファイルにアクセスする複数のクライアント計算機と、上記サーバ計算機と上記クライアント計算機とを接続するネットワークとからなるネットワークファイルサーバシステムであって、上記サーバ計算機は、ファイル等の情報を格納する記憶装置と、この記憶装置を制御する記憶装置制御手段と、上記クライアント計算機に送受信するバケットのプロトコル処理をし、上記クライアント計算機へ送信するバケットを格納する送信バッファ部と上記クライアント計算機から受信したバケットを格納する受信バッファ部と上記クライアント計算機からのリモートファイルシステム要求に基づいた処理を行うリモートファイル処理部とを設けた通信手段と、上記記憶装置を管理しこの記憶装置上のファイルを管理して、上記ファイルの上記記憶装置上の位置を問い合わせる要求に対して上記ファイルを格納しているディスクの位置を特定する情報を返す機能を持つローカルファイル制御手段と、上記クライアント計算機からのリモートファイルシステム要求の内容に基づき上記記憶装置上のファイルを読み書きするリモートファイル制御手段とを備え、上記リモートファイル処理部は、上記受信バッファが受信したバケットを解析し、このバケットに含まれている上記クライアント計算機が送信した上記ローカルファイル制御手段の管理するファイルに対する操作要求であるリモートファイルシステム要求を取り出し、このリモートファイルシステム要求が上記ローカルファイル制御手段の管理する上記記憶装置からのデータ転送を必要とする場合、上記ローカルファイル制御手段にそのファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記記憶装置制御手段に当該ファイルのデータを得られた位置情報に基づいて上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信バケットを作成して、上記クライアント計算機へリモートファイルシステム要求の返信を行い、上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記記憶装置制御手段にファイルのデータ

を得られた位置情報に基づいて上記送信バッファに転送することを要求し、このデータが上記送信バッファに格納されたら送信バケットを作成して上記クライアント計算機へリモートファイルシステム要求の返信を行い、上記リモートファイルシステム要求が上記記憶装置へのデータ転送を必要とする場合、上記ローカルファイル制御手段に当該ファイルが上記記憶装置のどの位置に存在するかを問い合わせる要求をし、上記受信バッファに格納しているデータを上記記憶装置制御手段に直接転送し、上記記憶装置制御手段が先に得た位置に基づき上記受信バッファに格納しているデータを上記記憶装置に書き込み、上記クライアント計算機へリモートファイルシステム要求に対する返信を行い、上記リモートファイルシステム要求が上記記憶装置からのデータ転送を必要としない場合は、上記リモートファイル制御手段にその要求を渡すように構成したことを特徴とする。

【0017】このような構成によれば、サーバ計算機上で動作するアプリケーション、オペレーティングシステム(OS)がディスク上のファイルを、ローカルファイルシステムを通して使用しつつ、ネットワークカードが、リモートファイルシステム要求を処理する際に、ディスクコントローラとネットワークカードの間で直接データを転送することで、ディスクコントローラとネットワークカード間の主記憶を用いたデータ転送の回数を減らすことで、高速な処理を行うことができる。

【0018】さらに、ネットワークカードが解釈したバケットのうち、ネットワークファイルシステム要求のみを、本発明で提案するリモートファイルシステム(サーバ計算機のプロセッサが実行する)に送る。このとき、要求のみをサーバ計算機の主記憶に送り、ファイルのデータ等をネットワークカードとディスクコントローラの間で直接DMA転送することにより、これまでよりデータ転送の回数を減らすことができる。

【0019】この構成に加えて、上記ローカルファイル制御手段が、要求のあったファイルを格納する記憶装置上の位置情報とこのファイルを特定する情報であるファイル位置情報を上記通信手段に通知する機能を備えれば、更に、高速にファイルのディスク位置情報を得ることができる。

【0020】また、この構成に加えて、サーバ計算機上で動作するアプリケーション、OSがディスク上のファイルを、ローカルファイルシステムを通して使用しつつ、ネットワークカードが、リモートファイルシステム要求を処理する際に、ローカルファイルシステムが管理する主記憶上のバッファキャッシュのデータをネットワークカードに直接転送することで、高速な処理を行うことができ、更に、リモートファイルシステム要求で発生したディスクとのデータのやりとりをバッファキャッシュに格納することで、次に同じファイルへの要求(サーバ計算機上のアプリケーションがローカルファイルシ

テムを通して、あるいは、クライアント計算機がリモートファイルシステム要求として)に対し、バッファキャッシュ上のデータを使用することができる。

【0021】

【発明の実施の形態】以下、本発明の一実施の形態について、図面を参照して説明する。

【0022】図1は、本実施形態に係わるネットワークファイルサーバシステムの構成を示したブロック図である。本実施形態の説明に必要な構成を示し、他の構成は省略している。

【0023】クライアント計算機200は、公衆通信網や有線、無線のLAN等のネットワーク300によってサーバ計算機100と接続されており、サーバ計算機100上のファイルにアクセスする。このファイルは、サーバ計算機上にインストールされた各種アプリケーション、またはその他のアプリケーション等に対応したファイルである。

【0024】プロセッサ101は、サーバ計算機100の各部の制御を行うものである。

【0025】システムバス102は、主記憶120等とプロセッサ101を接続し各種データ転送するものである。

【0026】I/Oバス103は、ネットワークカード130、ディスクコントローラ140、図示しないキーボード、マウス等の入力手段、ディスプレイ等のI/O機器とシステムバス102を接続し各種データを転送する。

【0027】主記憶120は、サーバ計算機100の主記憶であり、バッファキャッシュ121、送信バッファ122、受信バッファ123を備える。

【0028】リモートファイルシステム180、ローカルファイルシステム160、ネットワークカードドライバ170、ディスクドライバ150は、ソフトウェアであり、ディスク104等の記憶装置上に保存され、プロセッサ101により実行される。

【0029】リモートファイルシステム180は、ワーカーレッド181によりファイルシステム要求の内容に基づきサーバ計算機100上のファイルを読み書きする等の処理を行う。

【0030】ローカルファイルシステム160は、サーバ計算機100が備えるディスク104を管理し、ロック機能161とアンロック機能162を備える。ロック機能161とアンロック機能162の機能については後述する。

【0031】ネットワークカードドライバ170は、ネットワークカード130を制御するものである。

【0032】ディスクドライバ150は、ディスクコントローラ140によりディスク104を制御する。

【0033】サーバ計算機100は、クライアント計算機200とデータの送受信を行うためのプロトコル処理

を行うネットワークカード130を備える。このネットワークカード130は、サーバ計算機100とクライアント計算機200間で送受信するパケットのプロトコル処理を行った結果、そのパケットがクライアント計算機200からサーバ計算機100上のファイルを読み書き等するファイルシステム要求でない場合は、ネットワークカードドライバ170と連携して、通常のネットワークカードと同様の処理を行う。

【0034】また、ネットワークカード130は、上記送受信するパケットがファイルシステム要求の場合、サーバ計算機100上で動作するネットワークカードドライバ170と、リモートファイルシステム180上のワーカーズレッド181と協調して、ファイルシステム要求の内容に基づきサーバ計算機100上のファイルを読み書きする等の処理を行う。

【0035】上記ネットワークカード130は、以下のものから構成される。

【0036】受信パケットのプロトコル処理を実行し、その受信パケットがファイルシステム要求の場合には、その処理を行うリモートファイル処理手段136と、リモートファイル処理手段136が解析したパケットの内容を格納するファイル要求バッファ133、受信パケットが通常のパケットか、或いはファイルシステム要求か等を示す情報を格納するステータスレジスタ132、他の計算機から送信されたパケットをいったん格納する受信バッファ135と、送信すべきパケットを蓄積する送信バッファ134と転送要求レジスタ131とからなる。

【0037】転送要求レジスタ131は、データの格納元のアドレスとそこからデータ転送し、そのデータを格納する先のアドレスの情報が格納される。例えば、送信データがバッファキャッシュ121上にある場合は、そのデータのバッファキャッシュ121上のアドレスと、そのデータの転送先である送信バッファ134上のアドレスが格納される。受信データが受信バッファ135上にある場合、そのデータの受信バッファ135上のアドレスと、そのデータの転送先であるバッファキャッシュ121上のアドレスが格納される。

【0038】ステータスレジスタ132に格納するステータスを示す値は、ここでは次の通りとする。1はファイルシステムのリードライト要求受信を示し、2は通常パケット受信を示し、3はDMA転送完了を示し、そして、4は通常のパケット送信処理を示す。

【0039】ファイル要求バッファ133は、図2に示すようにリモートファイルシステム要求1330（リード、ライト、その他）、対象となるファイル名1331、そのオフセット1332、データ長1333、データアドレス1334から成るものである。

【0040】ローカルファイルシステム160は、サーバ計算機100が備えるディスクを管理する。即ち、デ

ィスク104上にファイルを格納し、ファイルの作成、リード、ライト、削除等の機能を提供する。主記憶120上にバッファキャッシュ121を作成し、ファイルのデータをキャッシュする。更に、以下の機能を備えるものである。

【0041】1. ファイル名、オフセット、データ長を入力として、そのファイルの該当する内容がバッファキャッシュ121にあるかどうかを判定する。バッファキャッシュ121にある場合には、そのバッファのアドレスを返す。

【0042】2. バッファのアドレスを入力として、該当するバッファへのファイル操作要求を禁止するロック機能161と、それを解除するアンロック機能162。

【0043】3. ファイル名、オフセット、データ長を入力として、該当するデータを格納しているディスクの位置（ディスクブロック）を返す。

【0044】ディスクコントローラ140は、ディスク要求処理手段142と、ディスク104から読み出した、あるいは、ディスク104へ書き込むデータをいったん格納するディスクキャッシュ141を備える。

【0045】サーバ計算機100が実行するネットワークカードドライバ170は、以下の機能を備える。

【0046】1. ネットワークカード130からの割り込みを処理する。

【0047】2. 通常のパケットの送受信を行う。

【0048】3. 主記憶120上に、送信バッファ122、受信バッファ123を確保する。

【0049】サーバ計算機100のプロセッサ101が実行するワーカーズレッド181は、リモートファイルシステム要求を受け、その処理を担当する。

【0050】リモートファイルシステム要求は、クライアント計算機200が、サーバ計算機100へ要求を送信するところから処理が始まる。以下、クライアント計算機200がリモートファイルシステム要求を出し、その要求をサーバ計算機100がどのように処理するかを図3のフローチャートにより説明する。

【0051】まず、ネットワークカード130のリモートファイル処理手段136は、クライアント計算機200からパケットの受信があると、以下の動作を行う。ここでは、通信にイーサネット、プロトコルにTCP/IPを用いている例を説明する。

【0052】ネットワークカード130宛てのパケットが、ネットワーク300上に発信された場合、そのパケットを受信バッファ135に蓄積する（ステップA1）。

【0053】受信バッファ135に蓄積したパケットを解析し、TCP/IPのプロトコル処理を行うパケット解析部1360がパケットのデータ部を解析し、処理できるプロトコルかどうか判定する（ステップA2）。

【0054】処理できないプロトコルの場合（ステップ

A2のNo)、ステップA8の処理を行う。

【0055】処理できるプロトコルの場合(ステップA2のYes)、パケット解析部1360によりプロトコル処理を行う(ステップA3)。

【0056】次に、リモートファイルシステム180への要求かどうかを判定する(ステップA4)。ここで、リモートファイルシステム180への要求は、あらかじめ定められたTCP/IPのポートに対し送信されるので、パケットのポート番号を監視すれば、容易に判定することができる。

【0057】リモートファイルシステム180への要求の場合は(ステップA4のYes)、ステータスレジスタ132を1(ファイルシステムのリードライト要求受信)にする(ステップA5)。

【0058】パケットからファイルシステム要求を取り出し、ファイル要求バッファ133に格納する。その要求は、ファイル操作、ファイル名、オフセット、データ長等からなる。さらに、要求がリードの場合は、ネットワークカード130上の送信バッファ134を確保して、そのアドレスをファイル要求バッファ133のデータアドレスに格納する。ライト要求の場合は、書き込むべきデータを格納したネットワークカード130上の受信バッファ135のアドレスをファイル要求バッファ133のデータアドレスに格納する(ステップA6)。

【0059】その後、プロセッサへ割り込みをかける(ステップA7)。

【0060】リモートファイルシステム180への要求でない場合、即ち、リード、ライト処理以外のファイルシステム要求の場合は(ステップA4のNo)、通常のパケットとして、ネットワークカードドライバ170があらかじめ確保した主記憶120上の受信バッファ135へパケットを転送する。ステータスレジスタを2(通常パケット受信)にして(ステップA8)、プロセッサへ割り込みをかける(ステップA7)。

【0061】データ転送が多い処理の場合は、本発明で提案する方法でデータをネットワークカード130、ディスクコントローラ140の間で直接やりとりする。本実施例の説明では、リード、ライト要求のみを扱っているが、その他にもデータ転送量が大きい要求があれば、同様に処理を行う。

【0062】次に、ネットワークカード130のDMA転送の処理動作を図4のフローチャートを参照して説明する。

【0063】転送要求レジスタ131にセット(格納)されている送信元アドレスから送信先アドレスへDMA転送でデータを送る(ステップB1)。

【0064】ステータスレジスタ132を3にする。3は、DMA転送完了を示す(ステップB2)。

【0065】プロセッサに割り込みをかける(ステップB3)。

【0066】次に、ネットワークカード130の送信の処理動作を図5のフローチャートにより説明する。

【0067】パケット解析部1360により、パケットが通常のパケット送信か否かを判定する(ステップC1)。

【0068】通常のパケット送信の場合(ステップC1のYes)、通常のパケット送信を実行する(ステップC2)。その後、ステータスレジスタ132を4にして(4は、通常のパケット送信処理を示す)(ステップC3)、プロセッサに割り込みをかける(ステップC4)。

【0069】通常のパケット送信でない場合(ステップC1のNo)、送信バッファのデータ134とファイル要求からクライアント計算機200への返答を作成する(ステップC5)。この作成したパケットについてパケット解析部1360によりプロトコル処理を行う(ステップC6)。パケットをネットワークに送信する(ステップC7)。

【0070】ネットワークカードドライバ170は、サーバ計算機100のプロセッサ101が実行する、ネットワークカード制御ソフトウェアである。その処理動作を図6のフローチャートにより説明する。

【0071】プロセッサが割り込みを受け、ネットワークカードドライバ170の割り込み処理ルーチンが呼び出される。

【0072】ネットワークカードドライバ割り込み処理ルーチンは、リモートファイルシステム180への要求か、通常のパケットかを、ネットワークカード130のステータスレジスタを調べることで判定する(ステップD1)。

【0073】リモートファイルシステム180への要求ではない場合、即ち、ステータスレジスタ132に格納された値が1以外の場合は(ステップD1のNo)、ネットワークカードドライバ170の割り込み処理ルーチンは、通常ネットワークカードドライバ170と同様の動作をする。

【0074】ここで、ステータスレジスタ132が2かどうかを判定する(ステップD4)。

【0075】ステータスレジスタ132に格納された値が2の場合(ステップD4のYes)、ネットワークカードドライバ170の割り込み処理ルーチンは、ネットワークカード130が転送したパケットのデータのうち、上位のプロトコル処理に不要なヘッダ(イーサネットフレーム情報)等を取り除き、プロトコルドライバ(TCP/IP等)に渡す(ステップD5)。以降は、通常のパケット受信と同じ処理になるので、ここでは説明を省略する。

【0076】リモートファイルシステム180への要求の場合、即ち、ステータスレジスタ132に格納された値が1の場合は(ステップD1のYes)、ネットワー

クカードドライバ170の割り込み処理ルーチンは、ネットワークカード130のファイル要求バッファ133から、ファイル要求を取り出す(ステップD2)。

【0077】このリモートファイルシステム要求を処理するワーカーズレッド181を起動し、要求を渡す(ステップD3)。

【0078】ステータスレジスタ132に格納された値が2以外の場合(但し、1は、既に除外されている)

(ステップD4のNo)、ステータスレジスタ132の値が3かどうかを判定する(ステップD6)。

【0079】ステータスレジスタ132の値が3の場合(ステップD6のYes)、DMA転送を要求しているスレッドを起動する(ステップD7)。

【0080】ステータスレジスタ132の値が3以外の場合(但し、1と2は既に除外されている)(ステップD6のNo)、ステータスレジスタ132の値が4かどうかを判定する(ステップD8)。

【0081】ステータスレジスタ132の値が4以外の場合(但し、1と2と3は既に除外されている)(ステップD8のNo)、処理は終了する。

【0082】ステータスレジスタ132の値が4の場合(ステップD8のYes)、通常のパケット送信完了処理を実行する(ステップD9)。

【0083】上記、図6のフローチャートの処理動作により、ネットワークカードドライバ170の割り込み処理はこれで完了し、以降、プロセッサが実行するワーカーズレッド181(リモートファイルシステム処理を行う)が、次に説明する一連の処理動作を実行する。

【0084】この処理動作を図7のフローチャートを参照して説明する。

【0085】ワーカーズレッド181は、渡されたファイル操作が、リードか、ライトか、その他の操作か、によって以下の処理を行う。

【0086】まず、ワーカーズレッド181は、渡されたファイル操作が、リード要求か否かを判定する(ステップE1)。

【0087】リード要求の場合(ステップE1のYes)、ワーカーズレッド181は、ファイル要求のファイル、オフセット、データ長から、リードしたい部分が主記憶120上のバッファキャッシュ121に存在するかどうかを、ローカルファイルシステム160の機能を使って調べる(ステップE2)。

【0088】バッファが存在する場合(ステップE2のYes)は、該当するバッファのアドレスを得ることができる。

【0089】バッファキャッシュ121に存在する場合は、以下の処理を行う。

【0090】該当するバッファをロックする操作を、ローカルファイルシステム160の機能を用いて行う(ステップE3)。この操作により、そのバッファに対する

新たな操作要求が来た場合には、その要求の処理を遅延する排他を行ったことになる。

【0091】ワーカーズレッド181は、転送要求レジスタ131にリードしたいバッファキャッシュ121上のデータのアドレスと、このデータを格納するネットワークカード130の送信バッファ134上のアドレスをセット(格納)し、この格納されたアドレスに基づき、バッファキャッシュ121のバッファの内容をネットワークカード130へDMA転送する要求を、ネットワークカード130のリモートファイル処理手段136に出し、その完了を待つ(ステップE4)。

【0092】ネットワークカード130のリモートファイル処理手段は、上記DMA要求を受けて、ネットワークカード130上の送信バッファ134にバッファを用意し、渡された主記憶120上のバッファのアドレスが指すデータをDMA転送する(ステップE5)。

【0093】ネットワークカード130のリモートファイル処理手段136は、DMA転送が終了すると、ステータスレジスタ132を3(DMA転送完了)にして、プロセッサに割り込みをかける(ステップE6)。

【0094】ネットワークカードドライバ170の割り込みルーチンが呼び出される。割り込みルーチンは、DMA転送完了の割り込みであることをステータスレジスタ132を読むことで認識する。ステップE5の処理で待っているワーカーズレッド181を起動し、割り込み処理を終了する(ステップE7)。

【0095】ワーカーズレッド181は、パケット送信要求をネットワークカード130のリモートファイル処理手段136に出す(ステップE8)。

【0096】リモートファイルシステム要求(リード要求)に対してのデータが、(2-3)の転送でネットワークカード130上の送信バッファに入っているため、後は、送信を実行すればよい。

【0097】ネットワークカード130のリモートファイル処理手段136は、送信バッファ134のデータを送信元(リード要求を出したクライアント計算機200)に送り返すためにイーサネットパケットのヘッダを作成し、送信を実行する(ステップE9)。

【0098】ワーカーズレッド181は、ローカルファイルシステム160の機能を用いて、主記憶120のバッファキャッシュ121のバッファへのロックを解除する(ステップE10)。以後、該当するバッファへの操作を他のスレッド、プロセス等が行うことができる。

【0099】バッファに存在しない場合は、以下の操作を行う(ステップE2のNo)。

【0100】ワーカーズレッド181は、ローカルファイルシステム160の機能を用いて、ファイルの読み出し操作で必要となるディスクブロックを調べる。ファイル名、オフセットの情報から、ディスク上のどの部分に要求されるデータが存在するかを調べる。ディスク10

4上の位置は、通常ディスクブロック番号等で管理されているので、ここではディスクブロック番号を得る（ステップE11）。

【0101】ワーカースレッド181は、ステップE11で得たディスクブロックを読み出し、ネットワークカードドライバ170の割り込みルーチンから渡されたネットワークカード130上の送信バッファ134にDMA転送を行うために、ディスクドライバ150を呼び出す（ステップE12）。

【0102】ディスクドライバ150は、その要求をディスクコントローラ140に伝え、完了を待つ（ステップE13）。

【0103】ディスクコントローラ140のディスク要求処理手段は、データをいったんディスク104から読み出し、そのデータをディスクキャッシュ141に格納する。そのキャッシュデータをステップE12で指示されたネットワークカード130の送信バッファ134に転送する（ステップE14）。

【0104】ディスクコントローラ140のディスク要求処理手段142は、プロセッサ101に割り込みをかける（ステップE15）。

【0105】ディスクドライバ150の割り込みルーチンは、ステップE13で、その完了を待っているワーカースレッド181を起動し、割り込み処理を終了する（ステップE16）。

【0106】ワーカースレッド181は、データ送信要求をネットワークカード130に出す（ステップE17）。

【0107】これは、クライアント計算機200から要求されたリモートファイルシステム要求（リード要求）のデータがネットワークカード130上に整ったので、ネットワークカード130に送信を指示することができるようになったからである。

【0108】ネットワークカード130のリモートファイル処理手段136は、送信バッファ134に入っているデータを使用して、リモートファイルシステム要求の返答を作成し、クライアント計算機200に送信する（ステップE18）。

【0109】ワーカースレッド181は、ローカルファイルシステム160の機能を用いて、ファイルの読み出し操作を出し、その完了を待つ（ステップE19）。

【0110】このファイル読み出し要求を処理する必要は、本来はない。なぜなら、クライアント計算機200から要求されたリード要求はすでに処理しているからである。しかし、ディスクから読み出したデータをサーバ計算機100のバッファキャッシュ121に格納しておき、次のアクセスがあった場合に、再利用できるようにしておくことで、キャッシュ効果が期待できるためにバッファキャッシュ121にデータを格納しておいてもよい。

【0111】その後、ワーカースレッド181の処理を終了する。

【0112】リード要求で無い場合（ステップE1のNo）、即ちライト要求の場合、ライト要求がライトスルー要求か否かを判定する（ステップE20）。

【0113】ライトスルー要求の場合（ステップE20のYes）、即ちディスクに即座にデータを書き込む要求の場合は、以下のような処理を行う。

【0114】ワーカースレッド181は、ファイル要求のファイル、オフセット、データ長から、ライトすべき部分が主記憶上のバッファキャッシュ121に存在するかどうかをローカルファイルシステム160の機能呼び出し調べる（ステップE21）。

【0115】ローカルファイルシステム160は、バッファが存在する場合（ステップE21のYes）、該当するバッファのアドレスを求め、ワーカースレッド181は、ローカルファイルシステム160の機能を用いて、そのバッファを破棄する（ステップE22）。これは、これから処理するライト要求によってデータが上書きされるので、主記憶120上のデータは、ディスク104に書き戻す必要がないためである。

【0116】ワーカースレッド181は、ローカルファイルシステム160の機能を用いて、ディスクブロックを調べ、ファイル要求のファイル、オフセット、データ長から、ライトするべきディスクブロックを得る（ステップE23）。

【0117】ワーカースレッド181は、ディスクドライバ150を用いて、ネットワークカード130の受信バッファ135上にあるデータをステップE23で得たディスクブロックへ書き込むように、要求を出し、その完了を待つ（ステップE24）。

【0118】ディスクコントローラ140のディスク要求処理手段142は、ディスク104の書き込みを終了すると、プロセッサ101に割り込みをかける（ステップE25）。

【0119】ディスクドライバ150の割り込み処理ルーチンは、ステップE24で完了を待っているワーカースレッド181を起こし、割り込み処理を終了する（ステップE26）。

【0120】ワーカースレッド181は、処理が完了したので、リモートファイルシステム要求に対する返答をクライアント計算機200に送信することを、ネットワークカード130に要求を出す（ステップE27）。

【0121】ネットワークカード130のリモートファイル処理手段136は、リモートファイルシステム要求の完了をクライアント計算機200に送信する（ステップE28）。

【0122】ワーカースレッド181は、ライト処理の指定を検査する（ステップE29）。

【0123】ライト処理がバッファキャッシュ121を

使用しないモードであれば(ステップE29のNo)、処理を終了する。

【0124】ライト処理がバッファを使用するモードの場合(ステップE29のYes)、バッファキャッシュ121にデータを入れておくために、ワークスレッド181は、ローカルファイルシステム160の機能を用いて、リード要求を処理し、その完了を待つ(ステップE30)。

【0125】このリード要求も本来は実行しなくてもよいものである。リモートファイルシステム要求を処理する場合、バッファキャッシュ121を使用する、しないは通常は、クライアント計算機200において意味があるものであるからである。

【0126】従って、ステップE30はなくてもよい。

【0127】その後、ワークスレッド181の処理を終了する。

【0128】ライト要求がライトスルーでない場合(ステップE20のNo)、以下の処理を行う。

【0129】ワークスレッド181は、ローカルファイルシステム160の機能を用い、ファイル要求のファイル、オフセット、データ長から、ライトすべき部分が主記憶上のバッファキャッシュ121に存在するかどうかを判定する(ステップE31)。

【0130】バッファが存在する場合(ステップE31のYes)、該当するバッファのアドレスを得ることができる。バッファが存在する場合は、以下の処理を行う。

【0131】ワークスレッド181は、ステップE31で得たバッファをロックする処理をローカルファイルシステム160の機能を用いて行う(ステップE32)。

【0132】ワークスレッド181は、受信バッファ135にあるデータのアドレスと、バッファキャッシュ121上のアドレスを転送要求レジスタ131にセットし、このアドレスに基づき、ネットワークカード130上の受信バッファ135にあるデータをステップE31で得たバッファキャッシュ121上のアドレスへDMA転送するように、ネットワークカード130のリモートファイル処理手段136に要求を出し、その完了を待つ(ステップE33)。

【0133】この転送により、バッファキャッシュ121上のデータを、リモートファイル要求(ライト要求)のデータに置き換えることになる。

【0134】ネットワークカード130のリモートファイル処理手段136は、DMA転送を完了したら、ステータスレジスタを3にセットして、プロセッサ101に割り込みをかける(ステップE34)。

【0135】ネットワークカードドライバ170の割り込みルーチンは、ステップE33で待つワークスレッド181を起こし、割り込みを完了する(ステップE3

5)。

【0136】ワークスレッド181、ロックしたバッファキャッシュ121上のバッファのロックを解除する(ステップE36)。

【0137】その後、ワークスレッド181の処理を終了する。

【0138】バッファがない場合(ステップE31のNo)、以下の処理を行う。

【0139】ワークスレッド181は、ローカルファイルシステム160の機能を用いて、バッファキャッシュ121上にバッファを確保し(ステップE37)、そのバッファをロックする(ステップE32)。

【0140】ステップE32以降は、バッファがある場合と同様の処理を続ける。

【0141】以上の様に、従来は、ネットワークカード130とディスクの間のデータ転送は、主記憶120にいったんデータを蓄積して行わなければならなかった。しかし、本発明によれば、リモートファイルシステム要求をネットワークカード130、ネットワークカードドライバ170、ディスクドライバ150とファイルシステムが協調して処理することで、ネットワークカード130とディスクコントローラ140の間で直接データをやりとりできる。また、そのデータのやりとりを主記憶120上のファイルシステムのバッファとの一貫性を保ちながら行っているため、サーバ計算機100上のローカルファイルシステム160がディスク104上のデータを操作する機能を維持している。

【0142】

【発明の効果】以上詳記したように本発明によれば、ネットワークカード、ディスクコントローラにあって、リモートファイル操作を処理するので、サーバ計算機のプロセッサは、リモートファイル操作を処理するためのプロトコル処理やファイル処理等を実行することがなくなり、それ以外の処理に資源、時間を割り当てることができ。

【0143】例えば、従来技術では、ライト処理の場合、ネットワークカードからデータを主記憶に転送し、更に主記憶からディスクへデータを転送するために、システムバス上に同一のデータが2回流れるものであったが、本発明によれば、ネットワークカードからディスクへI/Oバスのみを用いてデータを転送するので、システムバスを用いるデータの転送が必要ない。

【図面の簡単な説明】

【図1】本発明の一実施の形態に係わるネットワークファイルサーバシステムの構成を示すブロック図。

【図2】同実施の形態に係わるファイル要求バッファのデータ構成を示す図。

【図3】同実施の形態のネットワークカードの受信の際の処理動作を示すフローチャート。

【図4】同実施の形態に係わるネットワークカードのD

MA転送の際の処理動作を示すフローチャート。

【図5】同実施の形態に係わり、ネットワークカードの送信の際の処理動作を示すフローチャート。

【図6】同実施の形態に係わり、ネットワークカードドライバの割り込み処理ルーチンの処理動作を示すフローチャート。

【図7】同実施の形態に係わり、ワークスレッドの処理動作を示すフローチャート。

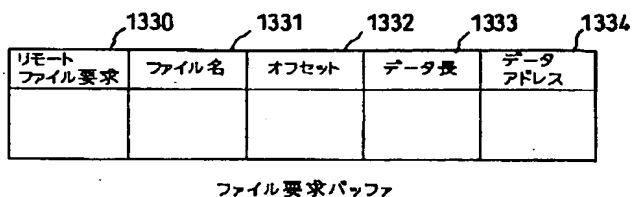
【図8】従来のネットワークファイルサーバシステムの構成を示す図。

【符号の説明】

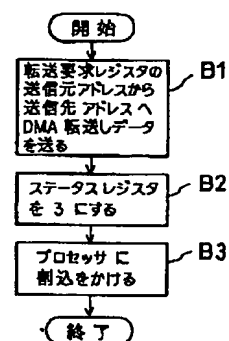
10…サーバ計算機
11…プロセッサ
12…システムバス
13…PCIバス
14…主記憶
15…バッファキャッシュ
16…ディスクコントローラ
17…ディスク
18…ネットワークカード
19…受信バッファ
20…ネットワークドライバ
21…プロトコルスタック
22…リモートファイルシステム
23…ローカルファイルシステム
24…ネットワーク
25…クライアント計算機
100…サーバ計算機
101…プロセッサ
102…システムバス
103…I/Oバス

104…ディスク
120…主記憶
121…バッファキャッシュ
122…送信バッファ
123…受信バッファ
130…ネットワークカード
131…転送要求レジスタ
132…ステータスレジスタ
133…ファイル要求バッファ
1330…リモートファイル要求
1331…ファイル名
1332…オフセット
1333…データ長
1334…データアドレス
134…送信バッファ
135…受信バッファ
136…リモートファイル処理手段
1360…バケット解析部
140…ディスクコントローラ
141…ディスクキャッシュ
142…ディスク要求処理手段
150…ディスクドライバ
160…ローカルファイルシステム
161…ロック機能
162…アンロック機能
170…ネットワークカードドライバ
180…リモートファイルシステム
181…ワークスレッド
200…クライアント計算機
300…ネットワーク
1330…リモートファイル要求

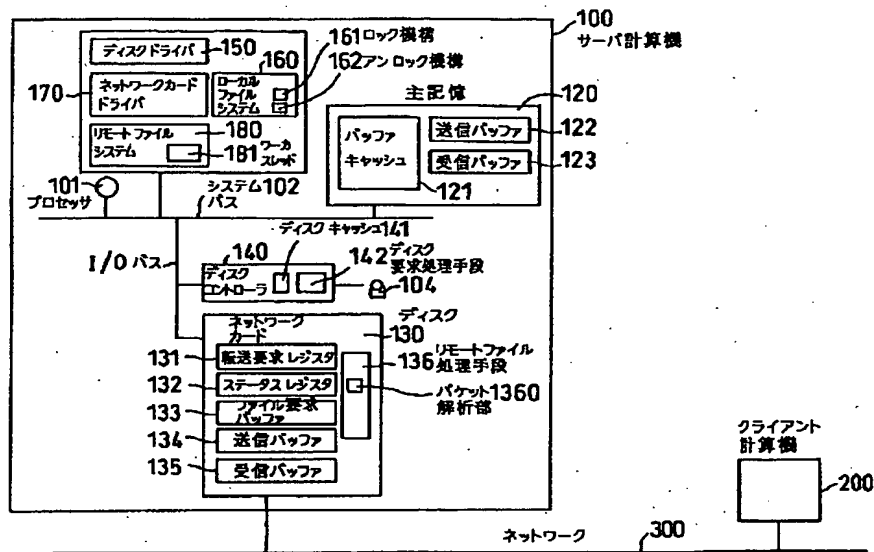
【図2】



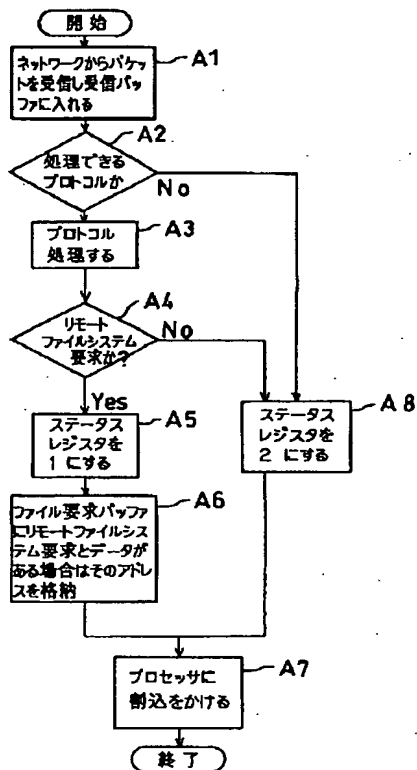
【図4】



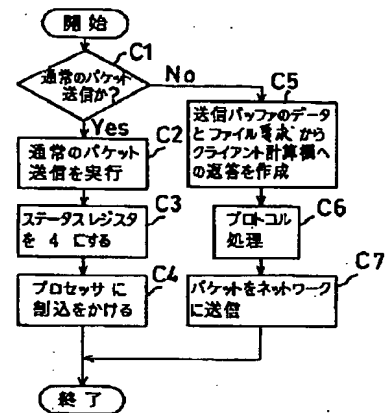
【図1】



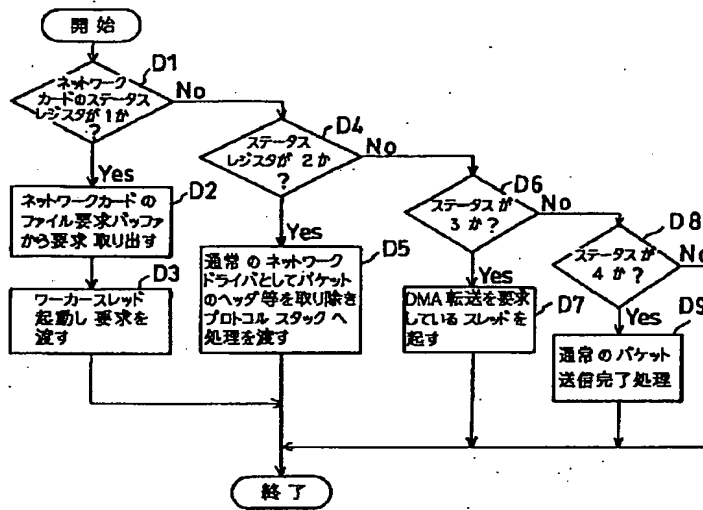
【図3】



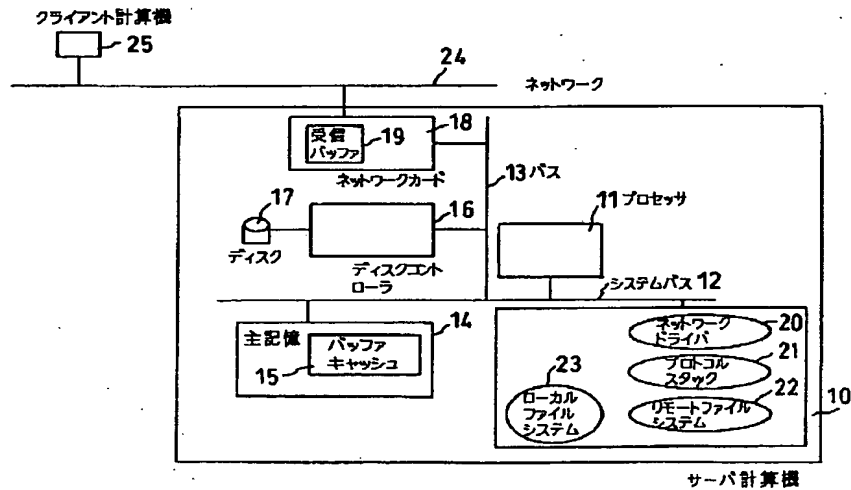
【図5】



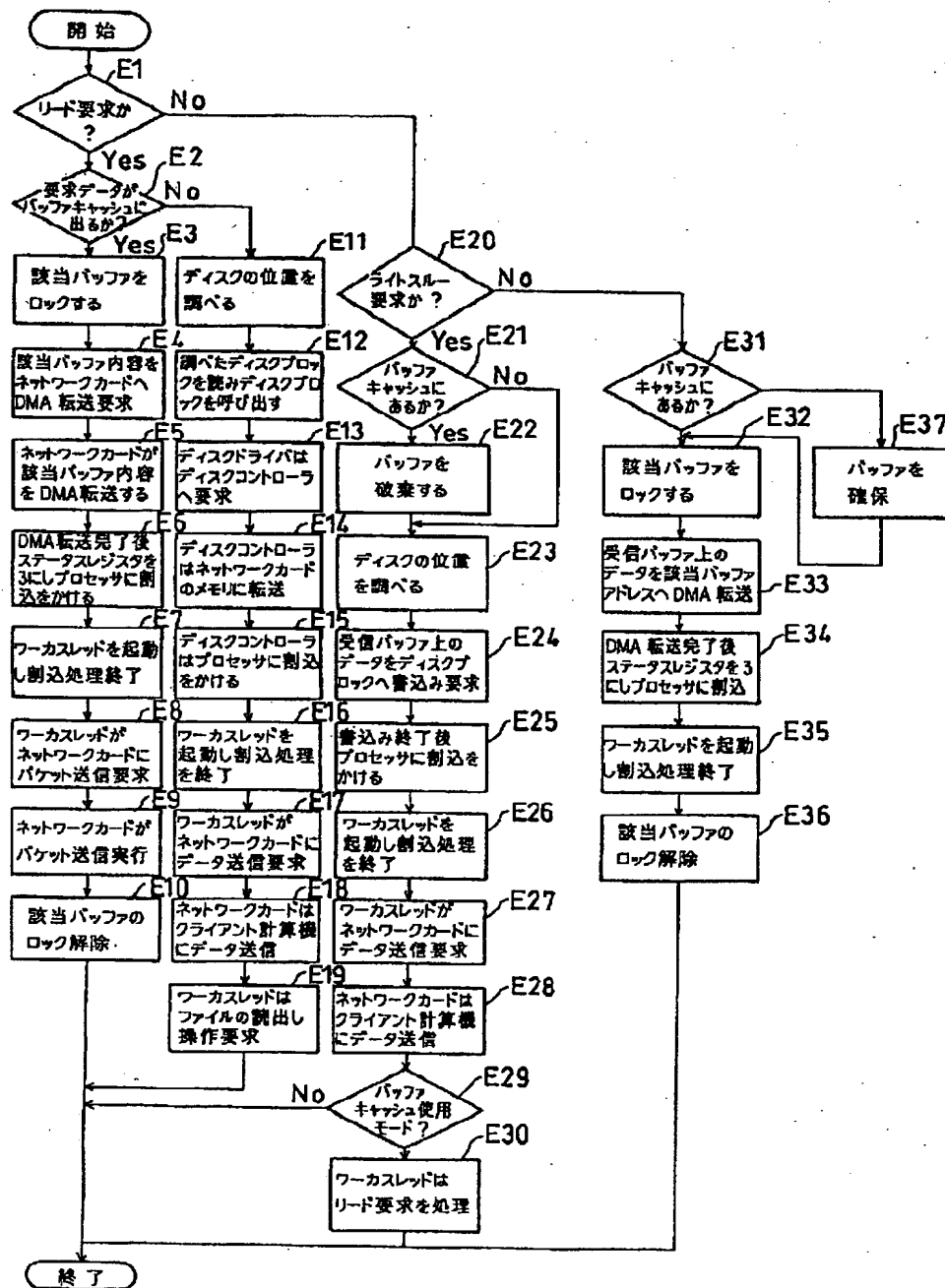
【図6】



【図8】



【図7】



PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-332782

(43)Date of publication of application : 02.12.1994

(51)Int.Cl.

G06F 12/00

G06F 15/16

G06F 15/16

(21)Application number : 06-050126

(71)Applicant : HITACHI LTD

HITACHI COMPUT ENG CORP LTD

(22)Date of filing : 22.03.1994

(72)Inventor : AKISAWA MITSURU

YAMASHITA YOJI

TADA KATSUMI

KAWAGUCHI HISAMITSU

KATO KANJI

KITO AKIRA

YAMADA HIDENORI

(30)Priority

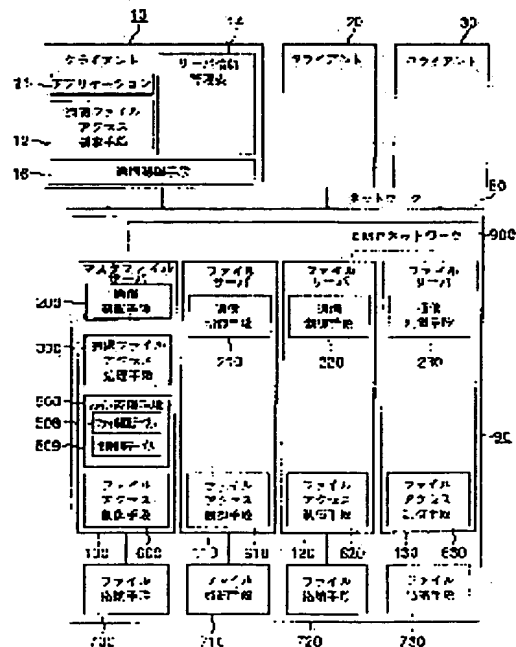
Priority number : 05 61602 Priority date : 22.03.1993 Priority country : JP

(54) FILE SERVER SYSTEM AND FILE ACCESS CONTROLLING METHOD THEREFOR

(57)Abstract:

PURPOSE: To prevent the throughput due to the centralization of access requests in a specified file server from plural clients, in a file server system where plural file servers accessing each file storage devices are arranged side by side via a network.

CONSTITUTION: The master file server 100 of file servers 100, 110, 120 and 130 composing a file server system 90 is provided with a file control means controlling files by using a load information table 508 measuring/controlling the load status of each file server and a file attribute table recording/controlling a file server in charge of the access every file block, selecting a file server where load is light at the time of writing a file, in particular and distributing the file access requests transmitted from client computers 10, 20 and 30 to the selected file server.



LEGAL STATUS

[Date of request for examination]

14.03.2001

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of

rejection]

[Date of requesting appeal against examiner's decision of
rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] In the file server system which shares the file which has two or more file servers **** (ed) on the network, and was distributed by the above-mentioned file server among two or more client computers A file storing means to store a file in each of two or more above-mentioned file servers. The first communications control means which performs communications control with other file servers through the above-mentioned network. A file access control means to receive a file access demand and to perform a file access to the above-mentioned file storing means is established. The second communications control means which performs communications control with the above-mentioned client computer to a specific file server among two or more above-mentioned file servers. A remote file-access-operation means to manage the communications protocol of the file access demand published from the above-mentioned client computer. A load information monitoring means to measure each load situation of two or more above-mentioned file servers. The file server which performs a file access from two or more above-mentioned file servers with reference to the load situation measured by the above-mentioned load information monitoring means is selected. When the selected file server is a self file server, a file access demand is published to the file access control means of a self file server. When the selected file servers are other file servers, the communications control means of the above 1st is minded. The file server system characterized by establishing further a file access demand distribution means to publish the file access demand which publishes a file access demand to the file access control means of the selected file server.

[Claim 2] The above-mentioned load information monitoring means is a file server system including the means which carries out counting of the unsettled number of file access demands in each of two or more above-mentioned file servers according to claim 1.

[Claim 3] The file server system according to claim 1 characterized by providing the following. The above-mentioned file access demand distribution means is a write-in read-out judging means for the file access demand published from the above-mentioned client computer to read in a write request, and to judge a demand. A file division arrangement means to select the file server which stores a file with reference to the load situation measured by the aforementioned load information monitoring means at the time of file writing. A file server scheduling means for access to select the file server made into the object of read-out from the file server in which it reads with reference to the load situation measured by the above-mentioned load information monitoring means at the time of file read-out, and the object file is stored.

[Claim 4] The above-mentioned file division arrangement means is a file server system according to claim 3 characterized by establishing a file division arrangement means to select at least two or more file servers which store the file for writing.

[Claim 5] It is the file-server system according to claim 3 characterized by for the above-mentioned file division arrangement means to select at least two or more file servers which store the file for writing, and for the above-mentioned file-server scheduling means for access to read a file server with a light load with reference to the load situation acquired by the aforementioned load information monitoring means among the file servers in which the file for read-out is stored, and to select as a target file server.

[Claim 6] It is the file server system according to claim 3 which the above-mentioned file division arrangement means generates the file attribute table showing the correspondence relation between this file and this file server in case the file server which stores a file is selected, and is carried out as the feature here where the file server in which the above-mentioned server scheduling means for access is read with reference to the above-mentioned file attribute table, and the target file is stored is specified.

[Claim 7] The file server system according to claim 1 characterized by preparing the communications control means of the above second, and the above-mentioned remote file-access-operation means or more in at least two of two or more above-mentioned file servers, respectively.

[Claim 8] The file server system according to claim 1 characterized by preparing the above-mentioned load information monitoring means and the above-mentioned file access demand distribution means or more in at least two of two or more above-mentioned file servers, respectively.

[Claim 9] The file server system according to claim 7 characterized by preparing the above-mentioned load information monitoring means and the above-mentioned file access demand distribution means or more in at least two of two or more above-mentioned file servers, respectively.

[Claim 10] It is the file server system according to claim 1 which the above-mentioned client computer is connected to the above-mentioned network, and is characterized by ** to which the communications control means of the above first achieves the function of the communications control of the above-mentioned client computer and the above-mentioned specific file server instead of the communications control means of the above second.

[Claim 11] It is the file server system according to claim 1 characterized by connecting the above-mentioned client computer to the 1st network, connecting two or more above-mentioned file servers to the 2nd network, and connecting the above 1st and the 2nd network with a bridge means to distribute the file access demand from the above-mentioned client computer to the above-mentioned specific file server.

[Claim 12] The communications control means of the above first is a file server system according to claim 1 characterized by performing communication with other file servers through a system bus.

[Claim 13] The above-mentioned system bus is a file server system according to claim 12 characterized by being the dedicated bus used only for communication between two or more above-mentioned file servers.

[Claim 14] In the file access control method in the file server system which shares the file which has two or more file servers ****(ed) on the network, and was distributed by two or more file servers of the above-mentioned file among two or more client computers The file access demand which measured each load information on two or more above-mentioned file servers, and was published through the above-mentioned network from the client computer in the **** case with a receptacle The file access control method containing the step of selecting the file server which performs a file access with reference to the above-mentioned load information, and distributing a file access demand to the above-mentioned selection file server.

[Claim 15] The step which measures the above-mentioned load information is the file access control method containing the step which carries out counting of each unsettled number of file access demands of two or more above-mentioned file servers according to claim 14.

[Claim 16] The step which selects the file server which performs the above-mentioned file access The file access demand published from the client computer reads in a write request, and judges a demand. At the time of file writing, the file server for file storing is selected with reference to the measured load situation. The file access control method according to claim 14 characterized by including the step of reading from the file server in which it reads with reference to the measured load situation at the time of file read-out, and the object file is stored, and selecting the target file server.

[Claim 17] The file access control method characterized by selecting at least two or more file servers which write in in the file access control method according to claim 16 at the step which

selects the file server for [above] file storing, and store an object file.

[Claim 18] The file access control method characterized by choosing and reading a file server with a light load based on the load information measured among two or more file servers in which it reads in the file access control method according to claim 16 at the step which selects the file server for [above] read-out, and the object file is stored, and considering as the target file server.

[Claim 19] In the file access control method according to claim 16, in case the file server which stores a file is selected, the file attribute table showing the correspondence relation between this file and this file server is generated, and the above-mentioned server scheduling means for access specifies the file server in which it reads with reference to the above-mentioned file attribute table, and the target file is stored. [Claim 20] In the file access control method according to claim 14, the step which selects the above-mentioned file server for a file access

The load information on each file server is measured by at least two or more file servers. The file access demand published from the client computer is received by the file server which measures the above-mentioned load information. The high-speed file access control method characterized by distributing a file access demand to the file server which selected the file server for a file access with reference to the above-mentioned load information, and was selected as a file server for a file access.

[Claim 21] The file access control method characterized by writing in a file server with few loads, selecting as an object file server when a file access is file writing, reading a file server with few loads from the file server in which the file is stored when a file access is file read-out in the file access control method according to claim 20 at the step which selects the above-mentioned file server for a file access, and selecting as an object file server.

[Translation done]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. *** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

- [0001] [Industrial Application] this invention is concerned with computing systems, such as a workstation and a server, and relates to the file server system especially in the computing system of multiprocessor composition which accesses the file stored in the secondary storage at high speed, and its file access control method.
- [0002] [Description of the Prior Art] In recent years, the network of a computer has been progressing. In connection with this, the need of the file server which manages collectively the file shared between computers is increasing. This is because a file system can be built by the low cost. That is, it is because two or more same files are copied and it is necessary to cease to possess them among two or more computers, since it becomes sharable [a file] among two or more computers by using a file server. It becomes possible to access as if it was the file stored in the client computer itself by usually carrying the network correspondence file system in the file server, and carrying a network correspondence file system access program also in the computer of a client side connected to the same network. Therefore, it becomes possible to access to the file which is accumulated on a file server from every client computer connected to the network, and is managed, and sharing of the file between two or more client computers is realized.
- [0003] Managing NFS and NIS (Hal Stern, O'Reilly & Associates, Inc, June 1991, p.113 to p.159) has the description about a network correspondence file system.
- [0004] However, when sharing the file in a file server using a network correspondence file system, a performance problem may arise. That is, in performing a file access simultaneously from many client computers, a load concentrates on a file server and the situation that an access result is not obtained immediately occurs.
- [0005] Therefore, when the load to a file server becomes excessive and the access throughput of a client computer falls, *** (ing) two or more sets of file servers on a network, and distributing the access demand from two or more client computer is performed. Each client computer manages the information on all the file servers currently installed on the network, and publishes an access demand to the file server in which the file for access exists with reference to this management information. When the file for access of each client computer is stored in the separate file server by this, it becomes possible to distribute a load and it can improve an access throughput.
- [0006] Although it is not client - and - server composition, one main processor manages two or more whole data files, and the above-mentioned system and a similar system are indicated by Japanese public presentation JP,04B -48352 [No. (1992)] in respect of if these data files are distributed by two or more sub processors.
- [0007] [Problem(s) to be Solved by the Invention] Though two or more sets of file servers are installed and a file is shared among many client computers as mentioned above, the following problems remain.
- [0008] Two or more client computers may give an access demand to one file server

simultaneously, an applicable file server serves as a bottleneck in that case, and the problem that a throughput will fall produces a client computer in order to access the file server in which a file exists completely regardless of the load situation of a file server. The degradation will become excessive if two or more directories where a client computer is the same and same files are accessed especially simultaneously. Even if this problem increases the installation number of the character top file server, it is not solvable.

[0009] The purpose of this invention is offering the file server system which can prevent generating of the bottleneck by concentration of access to a specific file server, and the fall of the throughput accompanying it, and its file access control method, even when two or more client computers access the same directory and the same file simultaneously in the network which *** (ed) two or more file servers, in order to share the file between many client computers.

[0010]

[Means for Solving the Problem] One feature of the file access control method of this invention In the file server system which shares the file which has two or more file servers *** (ed) on the network, and was distributed by two or more file servers among two or more client computers. The file access demand which measured each load information on two or more above-mentioned file servers, and was published through the above-mentioned network from the client computer in the *** case with a receptacle. The file server which performs a file access with reference to the above-mentioned load information is selected, and it is in the file access control method containing the step of distributing a file access demand to the above-mentioned selection file server.

[0011] When the file access demand from a client computer being a write request of a new file, speaking more concretely, the load information on two or more file servers is measured, the lightest file server of a load is selected, and a file write request is published to the file server. Or in producing the file of mirror composition, two or more file servers with a light load are selected, and it publishes a file write request to each of those file servers. Moreover, the file access demand from a client computer is a file read-out demand, when the file for read-out is carried out to mirror composition, the load information on two or more file servers which take charge of two or more multiple-files enclosure of each in which the file for read-out was stored is measured, one lightest file server of a load is selected, and a file read-out demand is published to the file server.

[0012] Measurement of the above-mentioned load information is performed by carrying out counting of each unsettled number of file access demands of two or more above-mentioned file servers. For this reason, at least one of two or more file servers is made into a master file server, and the load information monitoring means which records and updates each unsettled number of file access demands of two or more file servers at a load information table is prepared in this master file server. Moreover, a means to distribute the file access demand from a client computer using the file attribute table which records the correspondence relation of the file server which took charge of the writing of each file and its file, and the above-mentioned load information table is prepared in this master file server.

[0013] The typical composition of the file server system according to this invention In the file server system which shares the file which has two or more file servers *** (ed) on the network, and was distributed by the above-mentioned file server among two or more client computers. A file storing means to store a file in each of two or more above-mentioned file servers, The first communications control means which performs communications control with other file servers through the above-mentioned network, A file access control means to receive a file access demand and to perform a file access to the above-mentioned file storing means is established. On the other hand, a specific file server among two or more above-mentioned file servers at these In addition, the second communications control means which performs communications control with the above-mentioned client computer, A remote file-access-operation means to manage the communications protocol of the file access demand published from the above-mentioned client computer, A load information monitoring means to measure each load situation of two or more above-mentioned file servers, The file server which performs a file access from

two or more above-mentioned file servers with reference to the load situation measured by the above-mentioned load information monitoring means is selected. When the selected file server is a self file server, a file access demand is published to the file access control means of a self file server. It is the composition of having established further a file access demand distribution means to publish a file access demand to the file access control means of the selected file server through the communications control means of the above 1st when the selected file servers were other file servers.

[0014]

[Function] According to such a method and a system configuration, it can access to a file server with few file access loads. And since a file and its duplicate file are stored in two or more file servers, the access demand from a client computer to the same directory and the same file can also be distributed according to the load situation to two or more file servers. That is, even when ***((ing) two or more file servers, sharing a file among many client computers on a network and two or more directories and files with the same client computer are accessed simultaneously, generating of the bottleneck by concentration of access to a specific file server and the fall of the throughput accompanying it can be prevented, and access of the high throughput from a client computer can be realized.

[0015]

[Example] The composition of the example of this invention is explained with reference to drawing 1.

[0016] The file server system 90 constituted from a loose-coupling multiprocessor and the client computers 10, 20, and 30 are connected by Local Area Network 50. In each of the client computers 10, 20, and 30, if an application program 11 is performed and a file access demand occurs by this, a file access demand will be published from the remote file access demand generating means 12 in the file server system 90. Specifically, a file access demand is transmitted to the file server 100 which is specific one of four sets of the file servers 100, 110, 120, and 130 which constitute the file server system 90 through Local Area Network 50 from the communications control means 16. As for four sets of file servers 100, 110, 120, and 130, each takes charge of individually access of the file storing meanses 700, 710, 720, and 730. Therefore, the file access control meanses 600, 610, 620, and 630 are formed in file servers 100, 110, 120, and 130, respectively. Moreover, file servers 100, 110, 120, and 130 communicate mutually through the LCM network 900. For this reason, the communications control meanses 200, 210, 220, and 230 are formed in each file server.

[0017] In this example, the above-mentioned specific file server 100 is called master file server. The remote file-access-operation means 300 for the master file server 100 receiving the file access demand further published from the client computer and a file management means 500 to distribute the file access demand which managed and received distribution of a file so that the load of each file server might not incline greatly to each file server are formed. For distribution of this file management and a file access demand, the file attribute load information table 508 and 509 is used. The information on the master file server 100, i.e., the machine address of the master file server 100, is stored in the server information management table 14 of a client computer.

[0018] Drawing 2 is the block diagram showing the equipment configuration of the master file server 100. The master file server 100 includes the processor 101 mutually connected by the system bus 107, main memory 102, the network interface circuit 103, and the LCM network interface circuit 104. The remote file-access-operation program 301, file management 501, the file access control program 601, and a communication control program 201 are loaded to main memory 102 from the secondary storage which is not illustrated at the time of system starting, respectively, and the remote file-access-operation means 300 which this showed to drawing 1, the file management means 500, the file access control means 600, and the communications control means 200 are formed. A file storing means 700 for the master file server 100 to write in and to take charge of read-out is a magnetic disk unit as shown in drawing 2, and it is connected to a system bus 107. In addition, the file storing meanses 700 may be optical-magnetic disc equipment, an optical disk unit, and other secondary storages.

[0019] File servers 110, 120, and 130 other than master file server 100 of drawing 1 also have the composition almost same with being shown in drawing 2, respectively. However, a remote file access program and file management are not loaded to such main memory. Moreover, the network interface circuit 103 for connection to Local Area Network 50 is also unnecessary. [0020] Drawing 3 shows the program composition of the master file server 100. The network access program 205 which a communication control program 201 turns into to the interface of Local Area Network 50 and the master file server 100, The interprocessor communication drive-access program 206 used as the interface of the LCM network 900 and the master file server 100, The network communication protocol control program 207 which carries out protocol conversion and which is passed so that the remote file-access-operation program 301 can interpret the demand received from the network access program 205, It consists of interprocessor communication protocol control programs 208 which carry out protocol conversion of the access demand to other file servers interpreted by the file access control program 601 explained later, and are passed to the interprocessor communication drive-access program 206. The file access control program 601 receives the information about file enclosure and a file server from file management 501. When the self processor 101 accesses the file enclosure 700 which takes charge of an access control, the information about file enclosure is passed to the file enclosure access program 604. When other file servers 110-130 access the file enclosure 710-730 which takes charge of an access control, the information about file enclosure is passed to the interprocessor communication protocol control program 208. The file enclosure discernment program 603 which requests access from other file servers, The information about file enclosure is received from the file enclosure discernment program 603, and it consists of file enclosure access programs 604 which access the magnetic disk unit 700 which stores the target file.

[0021] File management 501 is explained also with reference to drawing 4 which shows still more detailed program composition. File management 501 changes into the information on a file server, the information on file enclosure, and the information on the file storing position in file enclosure the file access demand which managed the file attribute table 508 and the load information table 509, and was passed from the remote file-access-operation program 301 using these, and passes it to the file access control program 601. File server selection for [according to distribution and it of a file] access is processed. Therefore, the file management 501 The file access demand receptionist program 504 which the file access demand passed from the remote file-access-operation program 301 is received, and it reads in a write request, and distinguishes a demand. The file distribution program 502 which determines by which file server a file is written in at the time of file writing. The read-out demand scheduling program 503 which determines which file server is accessed at the time of read-out, It consists of load information monitoring programs 505 which measure the load situation of each file server by carrying out counting of the number of unsettled access demands of each file server. The information on the file storing position in the file server identifier corresponding to each file by which the file is stored in the file attribute table 508, a file enclosure identifier, and file enclosure is held. The number of unsettled access demands of each file server is held at the load information table 509.

[0022] Next, the 1st example of a file attribute table is shown in drawing 5. This example is an example in the case of performing distribution in a file unit, without having a duplicate, without dividing one file. A file attribute table consists of two fields, (1) file-attribute field and (2) disk-block index area. A file attribute field consists of each entry of a file size, file storing mode, a file access processor identifier, and a file storing device identifier. It is shown whether the file is stored in the file enclosure in which the file's being stored in the file enclosure to which the self file server which has managed a local, those with remote **, and the file attribute table takes charge of an access control in file storing mode, or other file servers take charge of an access control. A file access processor identifier shows the identifier of the file server which takes charge of the access control of file enclosure with which the file corresponding to a file attribute table is stored. A file storing device identifier shows the file enclosure in which the file is stored. The disk block index area consists of indexes which show the position within the file enclosure of each of a series of disk block which constitutes a file.

[0023] Next, the 2nd example of a file attribute table is shown in drawing 6. This example is an example in the case of distributing by dividing one file. A file attribute table consists of two fields, (1) file-attribute field and (2) disk-block index area, like the 1st example of drawing 5. However, the file attribute field existed for every disk block, and the storing place is specified. The example shown in drawing 6 shows that the 1st data block which constitutes a file exists in the position of the index of No. 100 of the disk unit of No. 1 where the 1st file server performs an access control. It is shown that the 3rd data block exists in the position of the index of No. 200 of the disk unit of No. 1 where, as for the 2nd data block, the 2nd file server performs an access control hereafter in the position of the index of No. 300 of the disk unit of No. 1 where the 3rd file server performs an access control.

[0024] On the other hand, the program composition of a file server 110 is as being shown in drawing 7. A communication control program 211 consists of an interprocessor communication drive-access program 212 which becomes the interface of transmission through the LCM network 900, i.e., an interface with the master file server 100, and an interprocessor communication protocol control program 213 which carries out protocol conversion and which is passed so that the file access control program 611 can interpret the access demand received from the interprocessor communication drive-access program 206. A file access control program interprets the access demand received from the interprocessor communication protocol control program 208, and consists of file enclosure access programs 612 which access the magnetic disk unit 710 which stores the target file. The program composition of file servers 120 and 130 is completely the same as that of drawing 7.

[0025] Next, operation of this example is explained using drawing 8.

[0026] If the processing demand which includes a file access demand or a file access by execution of an application program 11 occurs in either of the client computers 10, 20, and 30, the remote file access demand program 13 will be started, and a processing demand will be transmitted to the master file server 100 through Local Area Network 50. Communication through Local Area Network 50 is performed to a client computer and the master file server 100 using the communication control programs 17 and 201 carried, respectively. If a file access demand is sent to a file server 100, the remote file-access-operation program 301 of a file server 100 will be started. In the remote file-access-operation program 301, the received content is analyzed, the file access demand from a client computer is extracted, and a file-access-operation demand is sent to file management 501.

[0027] File management 501 operates, as shown in drawing 9. First, if judge whether the file access demand is writing and whether it is read-out, write in, come out, and it is, and the file distribution program 502 is started and read, comes out and it is after receiving the file access demand from the remote file-access-operation program 301 in the file access demand receptionist program 504, it will read and the demand scheduling program 503 will be started. In the file distribution program 502, the file attribute table to a write-in file is created, and next, a file server with few access demand unsettled numbers is determined with reference to the load information table 509 as a file server which stores a file. Moreover, in creating the duplicate of a file and storing in other file servers, with reference to the load information table 509, it determines a file server with few access demand unsettled numbers again as a file server which stores the duplicate of a file. It records on the file attribute table 508 by making into a server identifier the information on the file server which stores these files and a duplicate file, and the load information monitoring program 505 is started. In the read-out demand scheduling program 503, the file attribute table 508 to the file to read is gained, and the file server in which the applicable file is stored is deduced from there. When the file is stored not only in one file server but in the file server of others [duplicate / the], it determines which shall be read between a file main part and a duplicate with reference to the load information table 509, and the load information monitoring program 505 is started. In the load information monitoring program 505, by incrementing the access demand unsettled number to the file server for access, the monitoring of the load information is carried out, the information which shows which portion of the target file is accessed is sent to the file access control program 601, and the file access control program 601 is started.

[0028] The file access control program 601 operates according to the processing flow steps 651, 652, and 653 shown in drawing 8. It first judges whether the file enclosure discernment program 603 is access to the file enclosure 710-730 in which other file servers 110-130 take charge of an access control for whether it is access to the file enclosure 700 in which the information passed from file management 501 is analyzed, and the master file server 100 takes charge of an access control (Step 651). In being the former, the information about the file enclosure 700 is passed to the file enclosure access program 604 of the master file server 100, and it directs a file access. In response, the file enclosure access program 604 starts access to the file enclosure 700 (Step 652). In being the latter, it passes the information about file enclosure to the interprocessor communication protocol control program 208 in a communication control program 201, and the execution file server of a file access is specified, and a transfer of a demand is requested. The interprocessor communication protocol control program 208 is processed so that these information can be transmitted through the LCM network 900, and it is passed to the interprocessor communication drive-access program 206. The interprocessor communication drive-access program 206 sends out the received file access demand to the LCM network 900, and transmits it to the target file server. Here, it explains that a file server 110 is the target file server. In the file server 110 which is the destination of a file access demand, the interprocessor communication drive-access program 212 receives this demand, and the interprocessor communication protocol control program 213 is passed. The interprocessor communication protocol control program 213 will pass the information about file enclosure to the file enclosure access-control program 612 within the file access control program 611, if this recognizes that it is the file access demand sent from the master file server 100, i.e., other file servers. The file enclosure access program 612 is accessed to the file of the purpose of file enclosure.

[0029] Next, the method of storing of the file in an example is shown in drawing 10 and drawing 11. When the monitoring of the load of a file server is carried out by the load information monitoring program and it stores a file and its duplicate file in two file servers with the lightest load at the time of file writing, the combination of two file enclosure in which the mirror file of the same content is stored so that it may be shown drawing 10 becomes less fixed. On the other hand, it can always fix, it can have the pair of two file servers which store a file and its duplicate file, and the file data of the pair of file enclosure can also be completely considered as mirror composition like drawing 11. The pair of the file server which judges whether the load of which each file is light among two or more pairs of the file server together put also in this case, and stores servers may overlap mutually and may store the file data of the same content.

[0030] As mentioned above, according to this example, it becomes possible to read with the file distribution program 502 and to access to a file server with few file access loads by the demand scheduling program 503 by managing the number of the access demand which processing has not finished yet by the load information monitoring program 505, and carrying out the monitoring of the file access load to each file server. And since a file and its duplicate file are stored in two or more file servers, even if the access demand from a client computer to the same directory and the same file occurs simultaneously, it can distribute to two or more file servers. That is, even when ****(ing) two or more file servers, sharing a file among many client computers on a network and two or more directories and files with the same client computer are accessed simultaneously, generating of the bottleneck by concentration of access to a specific file server and the fall of the throughput accompanying it can be prevented, and access of the high throughput from a client computer can be realized.

[0031] In addition, although the case where the number of the file enclosure to which each processor performs file access control in the above-mentioned example was one respectively was shown, even if it is the composition which can connect and carry out the access control of two or more sets of the file enclosure to each file server, it is clear that the effect taken by this example and the same effect are acquired.

[0032] Furthermore, even if each program, such as a file enclosure discernment program shown by this example, file management, and a file access control program, consists of hardware, it is

clear that the effect taken by the above-mentioned this example and the same effect are acquired.

[0033] Another example of this invention is shown in drawing 12 - drawing 17, respectively. The example shown in drawing 12 is the composition of having prepared the remote file-access-operation program which existed only in the master file server 100 in other file servers 110, 120, and 130, respectively. Each file servers 100, 110, 120, and 130 are connected through the LCOMP network 900, respectively, and each-other data communication is performed. Furthermore, each file servers 100, 110, 120, and 130 are connected to Local Area Network 50. Therefore, it becomes possible to receive the processing demand which includes a file access demand or file access demand of a client computer through Local Area Network 50 in all file servers. For example, if the processing demand from a client computer communicates to a file server 110, the remote file-access-operation program 311 will interpret the content of communication, will extract a file access demand, and will start a communication control program 211. A communication control program 211 is carried out through the LCOMP network 900, and transmits a file access demand to the master file server 100. The master file server 100 determines the file server which stores a file by file management 501 like the example of drawing 1, or the file server which performs a readout.

[0034] It is the composition of having replaced the example with the composition in which file management exists in the example of drawing 1 only at the specific file server 100, and having prepared file management in drawing 13 at all file servers. Therefore, there is no distinction of a master and a slave among file servers 100, 110, 120, and 130. Furthermore, the storage region of each file enclosure 700, 710, 720, and 730 is quadrisectioned, respectively. Among [1-1] the divided fields, although 1-2, 1-3, and 1-4 were prepared in the 1st file server 100, they are a field which file management 501 manages. Moreover, although a field 2-1, 2-2, 2-3, and 2-4 were prepared in the 2nd file server 110, file management 511 prepared a field 3-1, 3-2, 3-3, and 3-4 in the 3rd file server 120 and file management 521 prepared a field 4-1, 4-2, 4-3, and 4-4 in the 4th file server 130, file management 531 is the field managed, respectively. That is, each file server has mutually the field which can manage self on the file enclosure in which other file servers take charge of an access control and to perform. The remote file-access-operation program 301 prepared only in the 1st file server 100 has the function to distribute the extracted file write request to the file management of each file server one by one. According to this example, distributed management of the file in a file server system can be carried out by four sets of file servers, and it becomes possible to perform file management to 4 sets to simultaneous parallel within one set of a file server system, therefore -- at the same time it acquires the effect of the rapid access of this invention -- a twist -- much more load distribution becomes possible and the effect that the degree of parallel of processing is raised and a file server system throughput can be improved is also acquired.

[0035] The example shown in drawing 14 is the composition of having prepared the remote file-access-operation program which existed in the chisel at the 1st file server 100 in all file servers, in the example shown in drawing 13. The effect of becoming possible to receive the processing demand including the file access demand or file access demand which minded LAN in all file servers is acquired at the same time the effect that a throughput can be improved by improvement in the load distribution by distributed management of a file and parallelism is acquired according to this example.

[0036] The example shown in drawing 15 is the file server system of the false loose-coupling multiprocessor composition which substitutes Local Area Network 50 for communication between file servers, without preparing a LCOMP network between each file server. The internal configuration of each file servers 100, 110, 120, and 130 is the same as that of the example shown in drawing 14. Also in the file server system of this example, it is clear that the effect acquired in each deformation example of the 1st example described above and the same effect are acquired.

[0037] The example shown in drawing 16 is an example which takes the composition connected to Local Area Network 50 through the bridge equipment 60 in which the false file server system of loose-coupling multiprocessor composition has the function which delivers a file access

demand to each file server, and realized the same function as an example to drawing 15's. Communication between each file servers 100, 110, 120, and 130 is performed using the network inside bridge equipment 60, and, in communication with an alien-machine system, is performed through bridge equipment. The monitoring of the load situation of each file server is carried out with bridge equipment 60, and if bridge equipment receives the file access demand from a client computer, it will select the file server which transmits a file access demand based on a load situation. Also in the file server system of this example, the effect acquired in the example's of drawing 14 described previously and the same effect are acquired.

[0038] The examples shown in drawing 17 are the example shown in drawing 14, and an example which realized the same function in the file server system of close coupling multiprocessor composition. The file server system of this example performs communication between file servers through this, using a system bus 80 as an interprocessor communication means. The network communication means connected to the system bus 80 performs communication with a client computer. Also in the file server system of this example, the effect acquired in each example described previously and the same effect are acquired.

[0039] [Effect of the Invention] According to this invention, it can access to a file server with few file access loads. And since a file and its duplicate file are stored in two or more file servers, the access demand from a client computer to the same directory and the same file can also be distributed according to the load situation to two or more file servers. Therefore, by the system which *** two or more file servers and shares a file among many client computers on a network, even when two or more directories and files with the same client computer are accessed simultaneously, generating of the bottleneck by concentration of access to a specific file server and the fall of the throughput accompanying it can be prevented, and access of the high throughput from a client computer can be realized.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

- [Drawing 1] It is the block diagram showing the whole example composition of this invention.
- [Drawing 2] It is the block diagram showing the detailed composition of the principal part of an example.
- [Drawing 3] It is the block diagram showing the program composition of the master file server of an example.
- [Drawing 4] It is the block diagram showing the composition of the file management of an example.
- [Drawing 5] It is the conceptual diagram showing an example of the file attribute table in an example.
- [Drawing 6] It is the conceptual diagram showing another example of the file attribute table in an example.
- [Drawing 7] It is the block diagram showing the program composition of other file servers of an example.
- [Drawing 8] It is the flow chart which shows the flow of the file access operation in an example.
- [Drawing 9] It is the flow chart which shows the flow of processing of the file management which can be set for it to be able to set in the example.
- [Drawing 10] It is the conceptual diagram showing an example of the file storing gestalt in an example.
- [Drawing 11] It is the conceptual diagram showing another example of the file storing gestalt in an example.
- [Drawing 12] It is the block diagram showing another example of this invention.
- [Drawing 13] It is the block diagram showing still more nearly another example of this invention.
- [Drawing 14] It is the block diagram showing still more nearly another example of this invention.
- [Drawing 15] It is the block diagram showing still more nearly another example of this invention.
- [Drawing 16] It is the block diagram showing still more nearly another example of this invention.
- [Drawing 17] It is the block diagram showing still more nearly another example of this invention.
- [Description of Notations]
- 10, 20, 30 -- A client computer, 11 -- Application program, 50 [-- Master file server,] -- A network, 90 -- A file server system, 100 110, 120, 130 -- A file server, 200, 210, 220, 230 -- Communications control means, 300 -- A remote file AKUSEKU processing means, 500 -- File management means, 508 [-- A file access control means, 700, 710, 720, 730 / -- A file storing means, 900 / -- LAMP network,] -- A file attribute table, 509 -- A load information table, 600, 610, 620, 630

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平6-332782

(43) 公開日 平成6年(1994)12月2日

(51) Int.Cl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 12/00	5 4 5 B	8944-5B		
15/16	3 7 0 M	7429-5L		
	3 8 0 Z	7429-5L		

審査請求 未請求 請求項の数21 O L (全 21 頁)

(21) 出願番号 特願平6-50126

(22) 出願日 平成6年(1994)3月22日

(31) 優先権主張番号 特願平5-61602

(32) 優先日 平5(1993)3月22日

(33) 優先権主張国 日本 (J P)

(71) 出願人 00005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(71) 出願人 000233011

日立コンピュータエンジニアリング株式会
社

神奈川県秦野市堀山下1番地

(72) 発明者 秋沢 充

東京都国分寺市東恋ヶ窪1丁目280番地

株式会社日立製作所中央研究所内

(74) 代理人 弁理士 小川 勝男

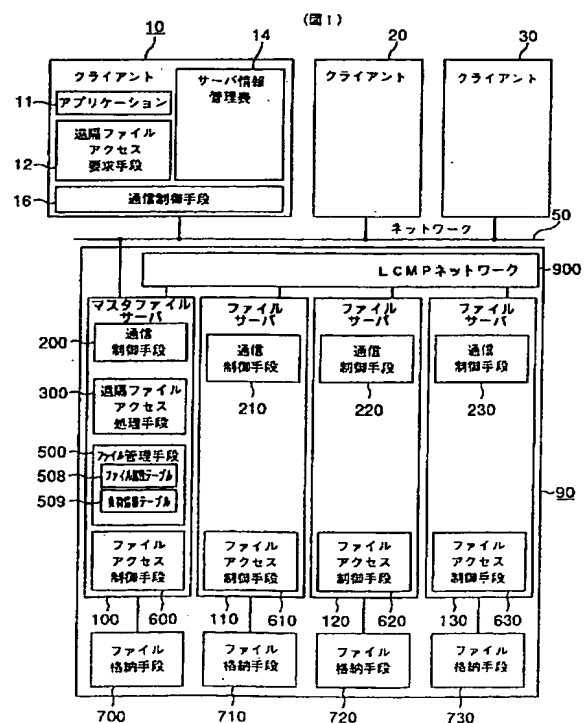
最終頁に続く

(54) 【発明の名称】 ファイルサーバシステム及びそのファイルアクセス制御方法

(57) 【要約】

【目的】 各々のファイル格納装置をアクセスする複数のファイルサーバがネットワークを介して並設されたファイルサーバシステムにおいて、複数のクライアントから特定のファイルサーバにアクセス要求が集中することによるスレープットの低下を防ぐ。

【構成】 ファイルサーバシステム90を構成するファイルサーバ100、110、120、130のうちのマスタファイルサーバ100には、各ファイルサーバの負荷状況を計測・管理する負荷情報テーブル508とファイルブロックごとのアクセス担当のファイルサーバを記録・管理するファイル属性テーブルとを用いてファイルを管理し、とくにファイル書き込み時には負荷の軽いファイルサーバを選定し、選定されたファイルサーバにクライアント計算機10、20、30から伝送されたファイルアクセス要求を分配するファイル管理手段を備える。



【特許請求の範囲】

【請求項1】 ネットワーク上に並接された複数のファイルサーバを有し、複数のクライアント計算機間で上記ファイルサーバに分散配置されたファイルを共有するファイルサーバシステムにおいて、

上記複数のファイルサーバの各々に、

ファイルを格納するファイル格納手段と、

上記ネットワークを介して他のファイルサーバとの通信制御を行う第一の通信制御手段と、

ファイルアクセス要求を受け付けて上記ファイル格納手段に対してファイルアクセスを行なうファイルアクセス制御手段を設け、

上記複数のファイルサーバのうち特定のファイルサーバに、

上記クライアント計算機との通信制御を行なう第二の通信制御手段と、

上記クライアント計算機から発行されたファイルアクセス要求の通信プロトコルを管理する遠隔ファイルアクセス処理手段と、

上記複数のファイルサーバの各々の負荷状況を計測する負荷情報モニタリング手段と、

上記負荷情報モニタリング手段によって計測した負荷状況を参照して上記複数のファイルサーバからファイルアクセスを行なうファイルサーバを選定し、選定されたファイルサーバが自己のファイルサーバであるときに自己のファイルサーバのファイルアクセス制御手段に対してファイルアクセス要求を発行し、選定されたファイルサーバが他のファイルサーバであるときに上記第1の通信制御手段を介してその選定されたファイルサーバのファイルアクセス制御手段に対してファイルアクセス要求を発行するファイルアクセス要求を発行するファイルアクセス要求配分手段を更に設けたことを特徴とするファイルサーバシステム。

【請求項2】 上記負荷情報モニタリング手段は、上記複数のファイルサーバの各々における未処理のファイルアクセス要求数を計数する手段を含む請求項1に記載のファイルサーバシステム。

【請求項3】 上記ファイルアクセス要求配分手段は、上記クライアント計算機から発行されたファイルアクセス要求が書き込み要求か読み出し要求かを判定する書き込み読み出し判定手段と、

ファイル書き込み時には前記負荷情報モニタリング手段によって計測した負荷状況を参照してファイルを格納するファイルサーバを選定するファイル分割配置手段と、ファイル読み出し時には上記負荷情報モニタリング手段によって計測した負荷状況を参照して読み出し対象ファイルが格納されているファイルサーバから読み出しの対象とするファイルサーバを選定するアクセス対象ファイルサーバ・スケジューリング手段を含むことを特徴とする請求項1に記載のファイルサーバシステム。

【請求項4】 上記ファイル分割配置手段は、書き込み対象のファイルを格納するファイルサーバを少なくとも二つ以上選定するファイル分割配置手段を設けたことを特徴とする請求項3に記載のファイルサーバシステム。

【請求項5】 上記ファイル分割配置手段は、書き込み対象のファイルを格納するファイルサーバを少なくとも二つ以上選定し、上記アクセス対象ファイルサーバ・スケジューリング手段は読み出し対象のファイルが格納されているファイルサーバのうち前記負荷情報モニタリング手段により取得された負荷状況を参照し負荷の軽いファイルサーバを読み出し対象のファイルサーバとして選定することを特徴とする請求項3に記載のファイルサーバシステム。

【請求項6】 上記ファイル分割配置手段は、ファイルを格納するファイルサーバを選定する際に、該ファイルと該ファイルサーバの対応関係を示すファイル属性テーブルを生成し、上記アクセス対象サーバスケジューリング手段は上記ファイル属性テーブルを参照して読み出し対象のファイルが格納されているファイルサーバを特定することを特徴とする請求項3に記載のファイルサーバシステム。

【請求項7】 上記第二の通信制御手段と上記遠隔ファイルアクセス処理手段とを上記複数のファイルサーバのうちの少なくとも二つ以上にそれぞれ設けたことを特徴とする請求項1に記載のファイルサーバシステム。

【請求項8】 上記負荷情報モニタリング手段と上記ファイルアクセス要求配分手段とを上記複数のファイルサーバのうちの少なくとも二つ以上にそれぞれ設けたことを特徴とする請求項1に記載のファイルサーバシステム。

【請求項9】 上記負荷情報モニタリング手段と上記ファイルアクセス要求配分手段とを上記複数のファイルサーバのうちの少なくとも二つ以上にそれぞれ設けたことを特徴とする請求項7に記載のファイルサーバシステム。

【請求項10】 上記クライアント計算機は上記ネットワークに接続され、上記クライアント計算機と上記特定のファイルサーバとの通信制御の機能は上記第二の通信制御手段の代りに上記第一の通信制御手段が果たすことを特徴とする請求項1に記載のファイルサーバシステム。

【請求項11】 上記クライアント計算機は第1のネットワークに接続され、上記複数のファイルサーバは第2のネットワークに接続され、上記第1、第2のネットワークは上記クライアント計算機からのファイルアクセス要求を上記特定のファイルサーバへ配分するブリッジ手段で接続されることを特徴とする請求項1に記載のファイルサーバシステム。

【請求項12】 上記第一の通信制御手段はシステムバスを介して他のファイルサーバとの通信を行うことを特徴とする請求項1に記載のファイルサーバシステム。

【請求項13】 上記システムバスは上記複数のファイルサーバ間の通信のみに用いる専用バスであることを特徴

とする請求項12に記載のファイルサーバシステム。

【請求項14】ネットワーク上に並接された複数のファイルサーバを有し、複数のクライアント計算機間で上記ファイルの複数のファイルサーバに分散配置されたファイルを共有するファイルサーバシステムにおけるファイルアクセス制御方法において、

上記複数のファイルサーバの各々の負荷情報を計測し、クライアント計算機から上記ネットワークを介して発行されたファイルアクセス要求が受け付けられた際に、上記負荷情報を参照してファイルアクセスを行うファイルサーバを選定し、

上記選定ファイルサーバに対してファイルアクセス要求を配分する、

とのステップを含むファイルアクセス制御方法。

【請求項15】上記の負荷情報を計測するステップは上記複数のファイルサーバの各々の未処理のファイルアクセス要求数を計数するステップを含む請求項14に記載のファイルアクセス制御方法。

【請求項16】上記ファイルアクセスを行なうファイルサーバを選定するステップは、

クライアント計算機から発行されたファイルアクセス要求が書き込み要求か読み出し要求かを判定し、

ファイル書き込み時には計測した負荷状況を参照してファイル格納対象のファイルサーバを選定し、

ファイル読み出し時には計測した負荷状況を参照して読み出し対象ファイルが格納されているファイルサーバから読み出し対象のファイルサーバを選定する、

とのステップを含むことを特徴とする請求項14に記載のファイルアクセス制御方法。

【請求項17】請求項16に記載のファイルアクセス制御方法において、上記のファイル格納対象のファイルサーバを選定するステップでは書き込み対象ファイルを格納するファイルサーバを少なくとも二つ以上選定することを特徴とするファイルアクセス制御方法。

【請求項18】請求項16に記載のファイルアクセス制御方法において、上記の読み出し対象のファイルサーバを選定するステップでは読み出し対象ファイルが格納されている複数のファイルサーバのうち計測した負荷情報に基づき負荷の軽いファイルサーバを選択して読み出し対象のファイルサーバとすることを特徴とするファイルアクセス制御方法。

【請求項19】請求項16に記載のファイルアクセス制御方法において、ファイルを格納するファイルサーバを選定する際に、該ファイルと該ファイルサーバの対応関係を示すファイル属性テーブルを生成し、上記アクセス対象サーバスケジューリング手段は上記ファイル属性テーブルを参照して読み出し対象のファイルが格納されているファイルサーバを特定する

【請求項20】請求項14に記載のファイルアクセス制御方法において、上記ファイルアクセス対象ファイルサ

ーバを選定するステップは、

少なくとも二つ以上のファイルサーバで各ファイルサーバの負荷情報を計測し、

クライアント計算機から発行されたファイルアクセス要求を上記負荷情報を計測するファイルサーバで受け取り、

上記負荷情報を参照してファイルアクセス対象ファイルサーバを選定し、

ファイルアクセス対象ファイルサーバとして選定したファイルサーバに対してファイルアクセス要求を配分することを特徴とする高速ファイルアクセス制御方法。

【請求項21】請求項20に記載のファイルアクセス制御方法において、上記ファイルアクセス対象ファイルサーバを選定するステップでは、ファイルアクセスがファイル書き込みである時には負荷の少ないファイルサーバを書き込み対象ファイルサーバとして選定し、ファイルアクセスがファイル読み出しである時にはファイルが格納されているファイルサーバから負荷の少ないファイルサーバを読み出し対象ファイルサーバとして選定することを特徴とするファイルアクセス制御方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明はワークステーションやサーバ等の計算機システムに関わり、特にマルチプロセッサ構成の計算機システムにおける、二次記憶装置に格納されたファイルを高速にアクセスするファイルサーバシステム及びそのファイルアクセス制御方法に関する。

【0002】

【従来の技術】近年、計算機のネットワーク化が進展してきている。これに伴い、計算機間で共有するファイルを一括して管理するファイルサーバの需要が高まっている。これは、低コストでファイルシステムを構築できるためである。すなわち、ファイルサーバを用いることによって複数の計算機間でファイルの共有が可能となるため、同一のファイルを複数の計算機間で複数コピーして所持しないですむようになるからである。ファイルサーバには通常ネットワーク対応ファイルシステムが搭載されており、同じネットワークに接続されたクライアント側の計算機にもネットワーク対応ファイルシステム・アクセスプログラムを搭載することによって、あたかもクライアント計算機自身に格納されているファイルであるかのようにアクセスすることが可能になる。そのため、ネットワークに接続されているどのクライアント計算機からも、ファイルサーバ上に蓄積、管理されているファイルに対してアクセスすることが可能となり、複数のクライアント計算機間でのファイルの共有が実現される。

【0003】ネットワーク対応ファイルシステムについての記述は、Managing NFS and NIS(Hal Stern, O'Reilly & Associates, Inc, June 1991, p.113～p.159)にある。

【0004】しかし、ファイルサーバ内のファイルをネットワーク対応ファイルシステムを用いて共有する場合には、性能上の問題が生じることがある。すなわち多数のクライアント計算機から同時にファイルアクセスを行なう場合には、ファイルサーバに負荷が集中し、直ちにアクセス結果が得られない状況が発生する。

【0005】そのため、ファイルサーバへの負荷が過大になりクライアント計算機のアクセス・スループットが低下する場合には、複数台のファイルサーバをネットワーク上に並接し複数クライアント計算機からのアクセス要求を分散することが行なわれている。各クライアント計算機は、ネットワーク上に設置されている全ファイルサーバの情報を管理し、この管理情報を参照してアクセス対象のファイルが存在するファイルサーバに対してアクセス要求を発行する。これにより、各クライアント計算機のアクセス対象ファイルが別々のファイルサーバに格納されている場合には、負荷を分散することが可能になり、アクセススループットを向上することができる。

【0006】クライアント・アンド・サーバ構成ではないが、一つのメインプロセッサが複数のデータファイルの全体の管理を行い、これらデータ・ファイルは複数のサブ・プロセッサに分散配置されているとの点で上記システムと類似するシステムが日本公開特許04-48352号(1992)に記載される。

【0007】

【発明が解決しようとする課題】上述したように複数台のファイルサーバを設置して多数のクライアント計算機間でファイルを共有したとしても、以下の問題が残る。

【0008】クライアント計算機はファイルサーバの負荷状況とは全く無関係にファイルが存在するファイルサーバをアクセスするため、複数のクライアント計算機が一つのファイルサーバに同時にアクセス要求を出すことがあり、その場合には該当ファイルサーバがボトルネックとなり、スループットが低下してしまうという問題が生じる。とくに、複数のクライアント計算機が同一のディレクトリや同一のファイルに同時にアクセスすると、その性能低下が甚だしくなる。この問題はその性格上ファイルサーバの設置台数を増やしても解決できるものではない。

【0009】本発明の目的は、多数のクライアント計算機間でのファイルの共有を行なうために複数のファイルサーバを並接したネットワークにおいて、複数のクライアント計算機が同一のディレクトリやファイルに同時にアクセスした場合でも、特定のファイルサーバへのアクセスの集中によるボトルネックの発生とそれに伴うスループットの低下を防ぐことができるファイルサーバシステムとそのファイルアクセス制御方法を提供することである。

【0010】

【課題を解決するための手段】本発明のファイルアクセ

ス制御方法の一つの特徴は、ネットワーク上に並接された複数のファイルサーバを有し、複数のクライアント計算機間で複数のファイルサーバに分散配置されたファイルを共有するファイルサーバシステムにおいて、上記複数のファイルサーバの各々の負荷情報を計測し、クライアント計算機から上記ネットワークを介して発行されたファイルアクセス要求が受け付けられた際に、上記負荷情報を参照してファイルアクセスを行うファイルサーバを選定し、上記選定ファイルサーバに対してファイルアクセス要求を配分する、とのステップを含むファイルアクセス制御方法にある。

【0011】より具体的にいえば、クライアント計算機からのファイルアクセス要求が新たなファイルの書き込み要求である場合には複数のファイルサーバの負荷情報を計測して負荷の最も軽いファイルサーバを選定し、そのファイルサーバにファイル書き込み要求を発行する。あるいは、ミラー構成のファイルを作製する場合には、負荷の軽い複数のファイルサーバを選定し、それらのファイルサーバそれぞれにファイル書き込み要求を発行する。また、クライアント計算機からのファイルアクセス要求がファイル読み出し要求であり、読み出し対象のファイルがミラー構成にされている場合には、読み出し対象のファイルが格納された複数の複数のファイル格納装置それぞれを受け持つ複数のファイルサーバの負荷情報を計測して負荷の最も軽いひとつのファイルサーバを選定し、そのファイルサーバにファイル読み出し要求を発行する。

【0012】上記の負荷情報の計測は、上記複数のファイルサーバの各々の未処理のファイルアクセス要求数を計数することにより行う。このために、複数のファイルサーバのうちの少なくとも一つをマスタ・ファイルサーバとし、このマスタ・ファイルサーバには、複数のファイルサーバの各々の未処理のファイルアクセス要求数を負荷情報テーブルに記録・更新する負荷情報モニタリング手段を設ける。また、このマスタ・ファイルサーバには、各ファイルとそのファイルの書き込みを受け持ったファイルサーバの対応関係を記録するファイル属性テーブル、および上記負荷情報テーブルを用いてクライアント計算機からのファイルアクセス要求を分配する手段を設ける。

【0013】本発明にしたがうファイルサーバシステムの代表的構成は、ネットワーク上に並接された複数のファイルサーバを有し、複数のクライアント計算機間で上記ファイルサーバに分散配置されたファイルを共有するファイルサーバシステムにおいて、上記複数のファイルサーバの各々には、ファイルを格納するファイル格納手段と、上記ネットワークを介して他のファイルサーバとの通信制御を行う第一の通信制御手段と、ファイルアクセス要求を受け付けて上記ファイル格納手段に対してファイルアクセスを行なうファイルアクセス制御手段を設

け、一方、上記複数のファイルサーバのうち特定のファイルサーバにはこれらに加えて、上記クライアント計算機との通信制御を行なう第二の通信制御手段と、上記クライアント計算機から発行されたファイルアクセス要求の通信プロトコルを管理する遠隔ファイルアクセス処理手段と、上記複数のファイルサーバの各々の負荷状況を計測する負荷情報モニタリング手段と、上記負荷情報モニタリング手段によって計測した負荷状況を参照して上記複数のファイルサーバからファイルアクセスを行なうファイルサーバを選定し、選定されたファイルサーバが自己のファイルサーバであるときに自己のファイルサーバのファイルアクセス制御手段に対してファイルアクセス要求を発行し、選定されたファイルサーバが他のファイルサーバであるときに上記第1の通信制御手段を介してその選定されたファイルサーバのファイルアクセス制御手段に対してファイルアクセス要求を発行するファイルアクセス要求配分手段を更に設けた、との構成である。

【0014】

【作用】このような方法及びシステム構成によれば、ファイルアクセス負荷の少ないファイルサーバへアクセスを行なうことができる。しかも、ファイルとその複製ファイルを複数のファイルサーバに格納するため同一のディレクトリやファイルに対するクライアント計算機からのアクセス要求も複数のファイルサーバにその負荷状況に応じて分散することができる。すなわち、ネットワーク上に複数のファイルサーバを並接し多数のクライアント計算機間でファイルの共有を行なう際に、複数のクライアント計算機が同一のディレクトリやファイルに同時にアクセスした場合でも、特定のファイルサーバへのアクセスの集中によるボトルネックの発生とそれに伴うスループットの低下を防ぐことができ、クライアント計算機からの高スループットのアクセスを実現できることになる。

【0015】

【実施例】本発明の実施例の構成を図1を参照して説明する。

【0016】疎結合マルチプロセッサで構成するファイルサーバシステム90と、クライアント計算器10、20、30とはローカルエリアネットワーク50により接続される。クライアント計算機10、20、30の各々ではアプリケーションプログラム11が実行され、これによってファイルアクセス要求が発生すると、遠隔ファイルアクセス要求発生手段12からファイルサーバシステム90へファイルアクセス要求が発行される。具体的には、ファイルアクセス要求は、通信制御手段16から、ファイルサーバシステム90を構成する4台のファイルサーバ100、110、120、130のうちの特定の一つであるファイルサーバ100へローカルエリアネットワーク50を介して伝送される。4台のファイル

サーバ100、110、120および130はそれぞれがファイル格納手段700、710、720および730のアクセスを個別に受け持つ。そのためにファイルサーバ100、110、120および130にはそれぞれファイルアクセス制御手段600、610、620および630が形成される。またファイルサーバ100、110、120および130はLCMPネットワーク900を介して互いに通信する。このために通信制御手段200、210、220、230がそれぞれのファイルサーバに形成される。

【0017】本実施例では、上記特定のファイルサーバ100をマスタファイルサーバと呼ぶ。マスタファイルサーバ100は、さらにクライアント計算機から発行されたファイルアクセス要求を受け付けるための遠隔ファイルアクセス処理手段300、および個々のファイルサーバの負荷が大きく偏らないようファイルの分散配置を管理し、且つ受け付けたファイルアクセス要求を個々のファイルサーバに振り分けるファイル管理手段500が形成される。このファイル管理およびファイルアクセス要求の振り分けのために、ファイル属性テーブル508および負荷情報テーブル509が用いられる。クライアント計算機のサーバ情報管理表14には、マスタファイルサーバ100の情報、つまりマスタファイルサーバ100のマシンアドレスが格納される。

【0018】図2はマスタファイルサーバ100の装置構成を示すブロック図である。マスタファイルサーバ100はシステムバス107で互いに接続されたプロセッサ101、主メモリ102、ネットワークインタフェース回路103及びLCMPネットワークインタフェース回路104を含む。システム立ち上げ時に、図示しない2次記憶装置から遠隔ファイルアクセス処理プログラム301、ファイル管理プログラム501、ファイルアクセス制御プログラム601及び通信制御プログラム201がそれぞれ主メモリ102にロードされ、これにより図1に示した遠隔ファイルアクセス処理手段300、ファイル管理手段500、ファイルアクセス制御手段600及び通信制御手段200が形成される。マスタファイルサーバ100が書き込み読み出しを受け持つファイル格納手段700は、図2に示す通り磁気ディスク装置であり、システムバス107に接続される。なおファイル格納手段700は光磁気ディスク装置や光ディスク装置、またその他の二次記憶装置であっても構わない。

【0019】図1のマスタファイルサーバ100以外のファイルサーバ110、120及び130も、図2に示すのと同様な構成をそれぞれ有する。但し、これらの主メモリには遠隔ファイルアクセスプログラム並びにファイル管理プログラムはロードされない。またローカルエリアネットワーク50への接続のためのネットワークインタフェース回路103も不要である。

【0020】図3はマスタファイルサーバ100のプロ

グラム構成を示す。通信制御プログラム201は、ローカルエリアネットワーク50とマスタファイルサーバ100とのインタフェースとなるネットワークアクセスプログラム205と、LCMPネットワーク900とマスタファイルサーバ100とのインタフェースとなるプロセッサ間通信装置アクセスプログラム206と、ネットワークアクセスプログラム205から受け取った要求を遠隔ファイルアクセス処理プログラム301が解釈できるようにプロトコル変換して渡すネットワーク通信プロトコル制御プログラム207と、後で説明するファイルアクセス制御プログラム601で解釈された他のファイルサーバへのアクセス要求をプロトコル変換してプロセッサ間通信装置アクセスプログラム206に渡すプロセッサ間通信プロトコル制御プログラム208から構成される。ファイルアクセス制御プログラム601は、ファイル管理プログラム501からファイル格納装置およびファイルサーバに関する情報を受け取り、自己のプロセッサ101がアクセス制御を受け持つファイル格納装置700にアクセスする場合にはファイル格納装置アクセスプログラム604へファイル格納装置に関する情報を渡し、他のファイルサーバ110~130がアクセス制御を受け持つファイル格納装置710~730にアクセスする場合にはプロセッサ間通信プロトコル制御プログラム208にファイル格納装置に関する情報を渡して他のファイルサーバにアクセスを依頼するファイル格納装置識別プログラム603と、ファイル格納装置識別プログラム603からファイル格納装置に関する情報を受け取り、目的のファイルを格納する磁気ディスク装置700をアクセスするファイル格納装置アクセスプログラム604とから構成される。

【0021】ファイル管理プログラム501については、さらに詳細なプログラム構成を示す図4をも参照して説明する。ファイル管理プログラム501は、ファイル属性テーブル508と負荷情報テーブル509を管理し、またこれらを用いて遠隔ファイルアクセス処理プログラム301から渡されたファイルアクセス要求をファイルサーバの情報、ファイル格納装置の情報およびファイル格納装置内のファイル格納位置の情報に変換してファイルアクセス制御プログラム601に渡す。ファイルの分散配置及びそれにしたがうアクセス対象のファイルサーバ選定の処理をおこなう。そのために、ファイル管理プログラム501は、遠隔ファイルアクセス処理プログラム301から渡されるファイルアクセス要求を受け付けて、それが書き込み要求か読み出し要求を判別するファイルアクセス要求受付プログラム504と、ファイル書き込み時にどのファイルサーバによりファイルを書き込むかを決定するファイル分散配置プログラム502と、読み出し時にどのファイルサーバにアクセスを行なうかを決定する読み出し要求スケジューリングプログラム503と、各ファイルサーバの未処理アクセス要求数

を計数することにより各ファイルサーバの負荷状況を計測する負荷情報モニタリングプログラム505とから構成される。ファイル属性テーブル508には、各ファイルに対応しそのファイルが格納されているファイルサーバ識別子とファイル格納装置識別子およびファイル格納装置内のファイル格納位置の情報が保持される。負荷情報テーブル509には各ファイルサーバの未処理アクセス要求数が保持される。

【0022】次に、ファイル属性テーブルの第1の例を図5に示す。この例は、ひとつのファイルを分割せずに、あるいは複製を持たずに、ファイル単位での分散配置を行う場合の例である。ファイル属性テーブルは、

(1) ファイル属性領域、(2) ディスクブロックインデックス領域の二つの領域から構成される。ファイル属性領域は、ファイルサイズ、ファイル格納モード、ファイルアクセス・プロセッサ識別子、ファイル格納デバイス識別子の各エントリからなる。ファイル格納モードにはローカルとリモートがあり、ファイル属性テーブルを管理している自己のファイルサーバがアクセス制御を受けもつファイル格納装置にファイルが格納されているのか、あるいは他のファイルサーバがアクセス制御を受けもつファイル格納装置にファイルが格納されているのかを示す。ファイルアクセス・プロセッサ識別子は、ファイル属性テーブルに対応するファイルが格納されているファイル格納装置のアクセス制御を受けもつファイルサーバの識別子を示す。ファイル格納デバイス識別子は、ファイルが格納されているファイル格納装置を示す。ディスクブロックインデックス領域は、ファイルを構成する一連の各ディスクブロックの、ファイル格納装置内での位置を示すインデックスから構成されている。

【0023】次に、ファイル属性テーブルの第2の例を図6に示す。この例は、ひとつのファイルを分割して分散配置を行う場合の例である。ファイル属性テーブルは、図5の第1の例と同様に(1)ファイル属性領域、(2)ディスクブロックインデックス領域の二つの領域から構成される。ただし、各ディスクブロックごとにファイル属性領域が存在し、その格納場所を指定している。図6に示す例では、ファイルを構成する第1のデータブロックは第1のファイルサーバがアクセス制御を行う1番のディスク装置のインデックス100番の位置に存在することを示している。以下、第2のデータブロックは第2のファイルサーバがアクセス制御を行う1番のディスク装置のインデックス200番の位置に、第3のデータブロックは第3のファイルサーバがアクセス制御を行う1番のディスク装置のインデックス300番の位置に存在することを示している。

【0024】一方、ファイルサーバ110のプログラム構成は図7に示すとおりである。通信制御プログラム211は、LCMPネットワーク900を介する伝送のインタフェース、つまりマスタ・ファイルサーバ100と

のインタフェースとなるプロセッサ間通信装置アクセスプログラム212と、プロセッサ間通信装置アクセスプログラム206から受け取ったアクセス要求をファイルアクセス制御プログラム611が解釈できるようにプロトコル変換して渡すプロセッサ間通信プロトコル制御プログラム213から構成される。ファイルアクセス制御プログラムは、プロセッサ間通信プロトコル制御プログラム208から受け取ったアクセス要求を解釈し、目的のファイルを格納する磁気ディスク装置710にアクセスするファイル格納装置アクセスプログラム612で構成される。ファイルサーバ120、130のプログラム構成も図7と全く同様である。

【0025】次に本実施例の動作について図8を用いて説明する。

【0026】クライアント計算機10、20、30のいずれかで、アプリケーションプログラム11の実行によりファイルアクセス要求またはファイルアクセスを含む処理要求が発生すると、遠隔ファイルアクセス要求プログラム13が起動され、処理要求はローカルエリアネットワーク50を介してマスタファイルサーバ100へ伝送される。ローカルエリアネットワーク50を介した通信はクライアント計算機とマスタファイルサーバ100にそれぞれ搭載された通信制御プログラム17及び201を用いて行なわれる。ファイルアクセス要求がファイルサーバ100に送られると、ファイルサーバ100の遠隔ファイルアクセス処理プログラム301が起動される。遠隔ファイルアクセス処理プログラム301では、受信した内容を解析してクライアント計算機からのファイルアクセス要求を抽出し、ファイル管理プログラム501にファイルアクセス処理要求を送る。

【0027】ファイル管理プログラム501は図9に示すように動作する。まず、ファイルアクセス要求受付プログラム504では、遠隔ファイルアクセス処理プログラム301からのファイルアクセス要求を受け付けた後、そのファイルアクセス要求が書き込みなのか読み出しなのかを判断し、書き込みであればファイル分散配置プログラム502を起動し、読み出しであれば読み出し要求スケジューリングプログラム503を起動する。ファイル分散配置プログラム502では、書き込みファイルに対するファイル属性テーブルを作成し、次に負荷情報テーブル509を参照してアクセス要求未処理数の少ないファイルサーバをファイルを格納するファイルサーバとして決定する。また、ファイルの複製を作成して他のファイルサーバに格納する場合には再び負荷情報テーブル509を参照してアクセス要求未処理数の少ないファイルサーバをファイルの複製を格納するファイルサーバとして決定する。これらファイルと複製ファイルを格納するファイルサーバの情報をサーバ識別子としてファイル属性テーブル508に記録し、負荷情報モニタリングプログラム505を起動する。読み出し要求スケジュー

リングプログラム503では、読み出すファイルに対するファイル属性テーブル508を獲得し、そこから該当ファイルが格納されているファイルサーバを割り出す。ファイルが一つのファイルサーバだけでなく、その複製が他のファイルサーバに格納されている場合には、負荷情報テーブル509を参照してファイル本体と複製のどちらを読み出すかを決定し、負荷情報モニタリングプログラム505を起動する。負荷情報モニタリングプログラム505では、アクセス対象のファイルサーバに対するアクセス要求未処理数をインクリメントすることによって負荷情報をモニタリングし、目的のファイルのどの部分をアクセスするのかわかる情報をファイルアクセス制御プログラム601に送り、ファイルアクセス制御プログラム601を起動する。

【0028】ファイルアクセス制御プログラム601は図8に示す処理フローステップ651、652、653にしたがって動作する。まずファイル格納装置識別プログラム603はファイル管理プログラム501から渡された情報を解析しマスタファイルサーバ100がアクセス制御を受け持つファイル格納装置700へのアクセスであるのか、他のファイルサーバ110～130がアクセス制御を受け持つファイル格納装置710～730へのアクセスであるのかを判断する(ステップ651)。前者である場合には、マスタファイルサーバ100のファイル格納装置アクセスプログラム604にファイル格納装置700に関する情報を渡してファイルアクセスを指示する。ファイル格納装置アクセスプログラム604はこれを受けてファイル格納装置700へのアクセスを開始する(ステップ652)。後者である場合には通信制御プログラム201の中のプロセッサ間通信プロトコル制御プログラム208にファイル格納装置に関する情報を渡し、ファイルアクセスの実行ファイルサーバを指定して要求の転送を依頼する。プロセッサ間通信プロトコル制御プログラム208はこれらの情報をLCMPネットワーク900を介して転送できるように加工して、プロセッサ間通信装置アクセスプログラム206に渡す。プロセッサ間通信装置アクセスプログラム206は、受け取ったファイルアクセス要求をLCMPネットワーク900に送り出して目的のファイルサーバへ転送する。ここではファイルサーバ110が目的のファイルサーバであるとして説明する。ファイルアクセス要求の転送先であるファイルサーバ110では、この要求をプロセッサ間通信装置アクセスプログラム212が受け取り、プロセッサ間通信プロトコル制御プログラム213に渡す。プロセッサ間通信プロトコル制御プログラム213は、これがマスタファイルサーバ100すなわち他のファイルサーバから送られたファイルアクセス要求であることを認識すると、ファイルアクセス制御プログラム611内のファイル格納装置アクセス制御プログラム612へファイル格納装置に関する情報を渡す。ファイル格納装

置アクセスプログラム612はファイル格納装置の目的のファイルに対してアクセスを行なう。

【0029】次に実施例におけるファイルの格納のしかたを図10、図11に示す。負荷情報モニタリングプログラムによってファイルサーバの負荷をモニタリングしておき、ファイル書き込み時に最も負荷の軽い二つのファイルサーバにてファイルとその複製ファイルを格納する場合には、図10示すように同一内容のミラーファイルが格納される二つのファイル格納装置の組合せは一定ではなくなる。一方、ファイルとその複製ファイルを格納する二つのファイルサーバのペアを常に固定し、もって図11の様にファイル格納装置のペアのファイルデータを完全にミラー構成とすることもできる。この場合も、組み合わされたファイルサーバの複数のペアの間でいずれのペアの負荷が軽いかを判定して各ファイルを格納するファイルサーバのペアを決定することができる。これらに代えて、すべてのファイルサーバが同一内容のファイルデータを互いに重複して格納する様に構成することもできる。

【0030】以上、本実施例によれば、負荷情報モニタリングプログラム505によってまだ処理が終わっていないアクセス要求の個数を管理し、各ファイルサーバに対するファイルアクセス負荷をモニタリングすることにより、ファイル分散配置プログラム502と読み出し要求スケジューリングプログラム503によってファイルアクセス負荷の少ないファイルサーバへアクセスを行なうことが可能となる。しかも、ファイルとその複製ファイルを複数のファイルサーバに格納するため同一のディレクトリやファイルに対するクライアント計算機からのアクセス要求が同時に発生しても複数のファイルサーバに分散することができるようになる。すなわち、ネットワーク上に複数のファイルサーバを並接し多数のクライアント計算機間でファイルの共有を行なう際に、複数のクライアント計算機が同一のディレクトリやファイルに同時にアクセスした場合でも、特定のファイルサーバへのアクセスの集中によるボトルネックの発生とそれに伴うスループットの低下を防ぐことができ、クライアント計算機からの高スループットのアクセスを実現できることになる。

【0031】なお、上記の実施例においては各プロセッサがファイルアクセス制御を行うファイル格納装置が各々1台の場合を示したが、各ファイルサーバに複数台のファイル格納装置を接続しアクセス制御できるような構成であっても、本実施例で示した効果と同様の効果が得られることは明らかである。

【0032】さらに、本実施例で示したファイル格納装置識別プログラム、ファイル管理プログラム、ファイルアクセス制御プログラム等の各プログラムがハードウェアで構成されていても、上述の本実施例で示した効果と同様の効果が得られることは明らかである。

【0033】本発明の別の実施例を図12～図17にそれぞれ示す。図12に示す実施例はマスタファイルサーバ100のみに存在した遠隔ファイルアクセス処理プログラムを他のファイルサーバ110、120、130にもそれぞれ設けた構成である。各ファイルサーバ100、110、120、130はそれぞれLCMPネットワーク900を介して接続され、互いデータ通信が行われる。さらに各ファイルサーバ100、110、120、130はローカルエリアネットワーク50に接続される。したがってすべてのファイルサーバにおいてローカルエリアネットワーク50を介してクライアント計算機のファイルアクセス要求またはファイルアクセス要求を含む処理要求を受け付けることが可能となる。例えば、ファイルサーバ110にクライアント計算機からの処理要求が通信されると、遠隔ファイルアクセス処理プログラム311は通信内容を解釈してファイルアクセス要求を抽出し、通信制御プログラム211を起動する。通信制御プログラム211はLCMPネットワーク900を介してしてファイルアクセス要求をマスタファイルサーバ100に伝送する。マスタファイルサーバ100は、図1の実施例と同様にファイル管理プログラム501によりファイルを格納するファイルサーバ、もしくは読みだしを行うファイルサーバを決定する。

【0034】図13に実施例は、図1の実施例において特定のファイルサーバ100のみにファイル管理プログラムが存在する構成に代えて、ファイル管理プログラムをすべてのファイルサーバに設けた構成である。したがって、ファイルサーバ100、110、120、130の間には、マスタ、スレーブの区別はない。さらに、各ファイル格納装置700、710、720、730の記憶領域は、それぞれ4分割される。分割された領域のうち1-1、1-2、1-3、1-4は第1のファイルサーバ100に設けたがファイル管理プログラム501が管理する領域である。また、領域2-1、2-2、2-3、2-4は第2のファイルサーバ110に設けたがファイル管理プログラム511が、領域3-1、3-2、3-3、3-4は第3のファイルサーバ120に設けたがファイル管理プログラム521が領域4-1、4-2、4-3、4-4は第4のファイルサーバ130に設けたがファイル管理プログラム531がそれぞれ管理する領域である。つまり、各ファイルサーバは、互いに他のファイルサーバがアクセス制御を受け持つ行うファイル格納装置上に自身が管理可能な領域をもつ。第1のファイルサーバ100にのみ設けられた遠隔ファイルアクセス処理プログラム301は、抽出したファイル書き込み要求を各ファイルサーバのファイル管理プログラムに順次振り分ける機能を有する。本実施例によれば、ファイルサーバシステム内のファイルを4台のファイルサーバで分散管理することができ、1台のファイルサーバシステム内で4組までのファイル管理を同時並列に実行す

ることが可能となる。したがって本発明の高速アクセスの効果を得ると同時に、より一層の負荷分散が可能となり、処理の並列度を上げてファイルサーバシステムのスループットを向上できるという効果も得られる。

【0035】図14に示す実施例は図13に示した実施例において、第1のファイルサーバ100に、のみに存在した遠隔ファイルアクセス処理プログラムをすべてのファイルサーバに設けた構成である。本実施例によれば、ファイルの分散管理による負荷分散と並列性の向上によりスループットを向上できるという効果が得られると同時に、すべてのファイルサーバにおいてLANを介したファイルアクセス要求またはファイルアクセス要求を含む処理要求を受け付けることが可能となるという効果が得られる。

【0036】図15に示す実施例は、各ファイルサーバ間にL CMPネットワークを設けずに、ファイルサーバ間の通信をローカルエリアネットワーク50で代用する擬似的な疎結合マルチプロセッサ構成のファイルサーバシステムである。各ファイルサーバ100、110、120、130の内部構成は図14に示した実施例のと同様である。本実施例のファイルサーバシステムにおいても、以上述べた第1の実施例の各変形実施例で得られる効果と同様の効果が得られることが明らかである。

【0037】図16に示す実施例は、擬似的な疎結合マルチプロセッサ構成のファイルサーバシステムが、ファイルアクセス要求を各ファイルサーバへ配送する機能を持つブリッジ装置60を介してローカルエリアネットワーク50へ接続される構成をとるものであり、図15のに実施例と同様の機能を実現した実施例である。各ファイルサーバ100、110、120、130の間の通信はブリッジ装置60の内部のネットワークを用いて行い、他の計算機システムとの通信の場合にブリッジ装置を介して行う。各ファイルサーバの負荷状況はブリッジ装置60でモニタリングし、クライアント計算機からのファイルアクセス要求をブリッジ装置が受信すると負荷状況をもとにファイルアクセス要求を送信するファイルサーバを選定する。本実施例のファイルサーバシステムにおいても、先に述べた図14の実施例の得られる効果と同様の効果が得られる。

【0038】図17に示す実施例は密結合マルチプロセッサ構成のファイルサーバシステムにおいて、図14に示した実施例と同様の機能を実現した実施例である。本実施例のファイルサーバシステムはプロセッサ間通信手段としてシステムバス80を用い、これを介してファイルサーバ間の通信を行うものである。クライアント計算機との通信はシステムバス80に接続されたネットワーク通信手段により行う。本実施例のファイルサーバシステムにおいても、先に述べた各実施例で得られる効果と同様の効果が得られる。

【0039】

【発明の効果】本発明によれば、ファイルアクセス負荷の少ないファイルサーバへアクセスを行なうことができる。しかも、ファイルとその複製ファイルを複数のファイルサーバに格納するため同一のディレクトリやファイルに対するクライアント計算機からのアクセス要求も複数のファイルサーバにその負荷状況に応じて分散することができる。したがって、ネットワーク上に複数のファイルサーバを並接し多数のクライアント計算機間でファイルの共有を行なうシステムで、複数のクライアント計算機が同一のディレクトリやファイルに同時にアクセスした場合でも、特定のファイルサーバへのアクセスの集中によるボトルネックの発生とそれに伴うスループットの低下を防ぐことができ、クライアント計算機からの高スループットのアクセスを実現できる。

【図面の簡単な説明】

【図1】本発明の実施例の全体構成を示すブロック図である。

【図2】実施例の主要部の詳細構成を示すブロック図である。

【図3】実施例のマスタファイルサーバのプログラム構成を示すブロック図である。

【図4】実施例のファイル管理プログラムの構成を示すブロック図である。

【図5】実施例におけるファイル属性テーブルの一例を示す概念図である。

【図6】実施例におけるファイル属性テーブルの別の例を示す概念図である。

【図7】実施例の他のファイルサーバのプログラム構成を示すブロック図である。

【図8】実施例におけるファイルアクセス処理の流れを示すフローチャートである。

【図9】実施例におけるにおけるファイル管理プログラムの処理の流れを示すフローチャートである。

【図10】実施例におけるファイル格納形態の一例を示す概念図である。

【図11】実施例におけるファイル格納形態の別の例を示す概念図である。

【図12】本発明の別の実施例を示すブロック図である。

【図13】本発明のさらに別の実施例を示すブロック図である。

【図14】本発明のさらに別の実施例を示すブロック図である。

【図15】本発明のさらに別の実施例を示すブロック図である。

【図16】本発明のさらに別の実施例を示すブロック図である。

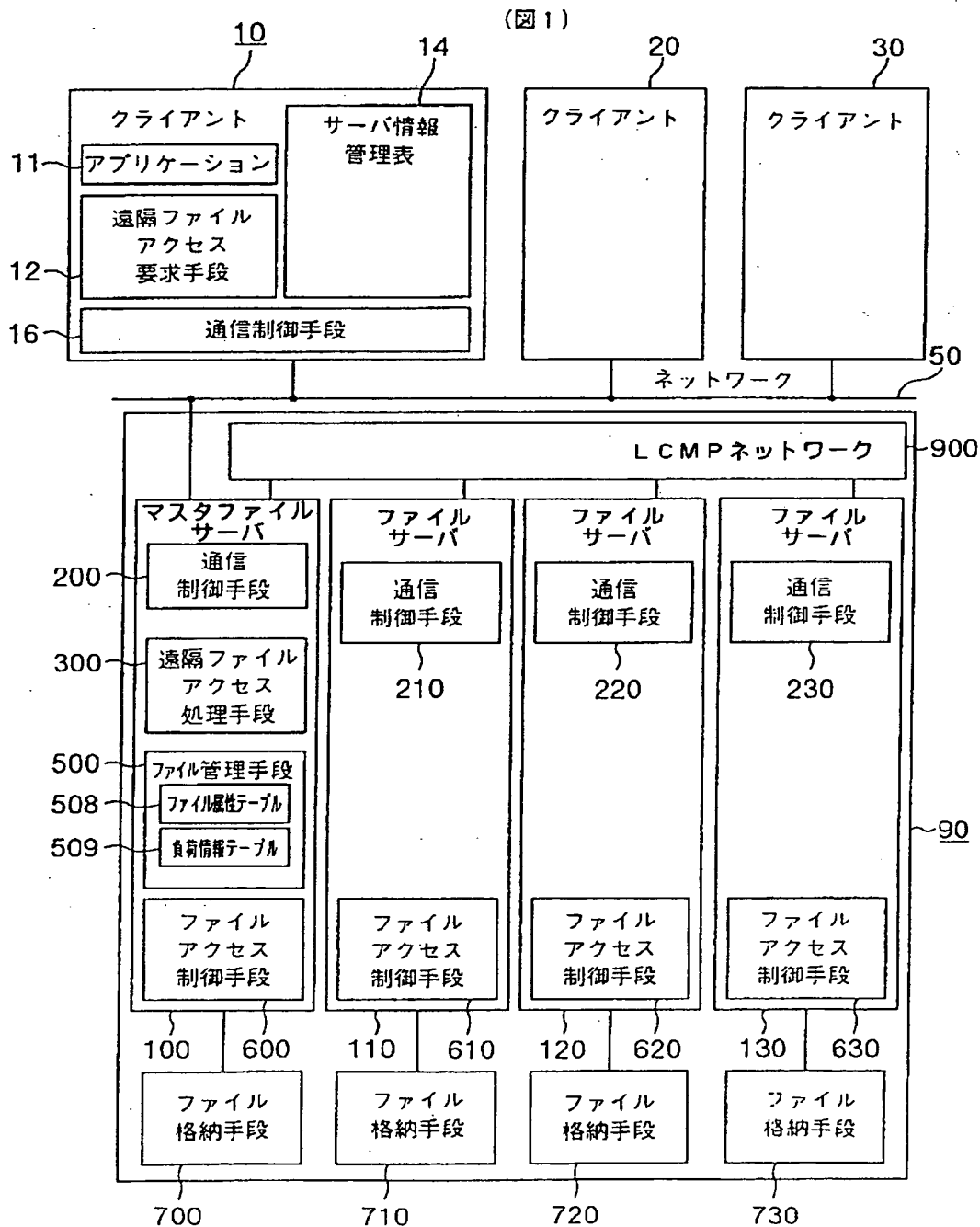
【図17】本発明のさらに別の実施例を示すブロック図である。

【符号の説明】

10、20、30…クライアント計算機、11…アプリケーションプログラム、50…ネットワーク、90…ファイルサーバシステム、100…マスタ・ファイルサーバ、110、120、130…ファイルサーバ、200、210、220、230…通信制御手段、300…

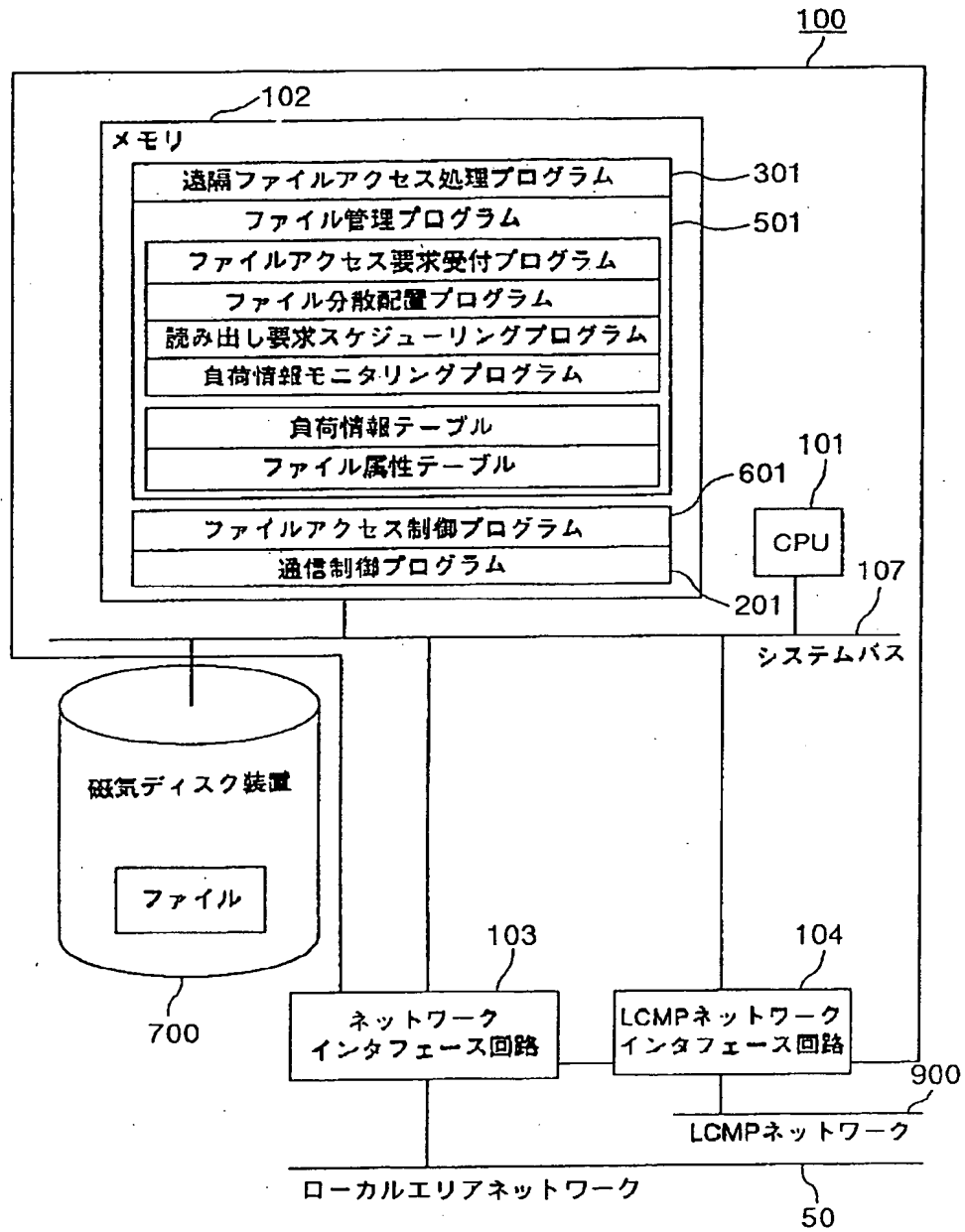
遠隔ファイルアクセス処理手段、500…ファイル管理手段、508…ファイル属性テーブル、509…負荷情報テーブル、600、610、620、630…ファイルアクセス制御手段、700、710、720、730…ファイル格納手段、900…LCMPネットワーク。

【図1】



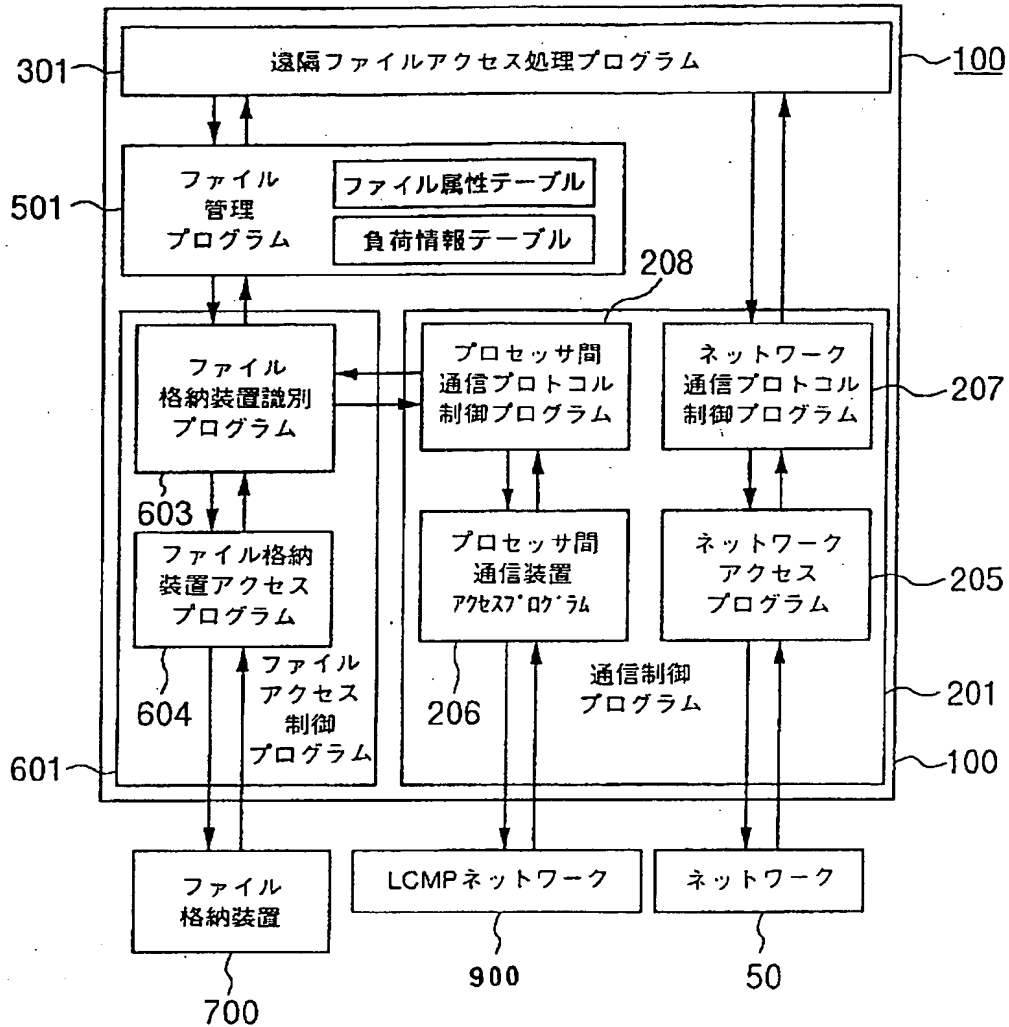
【図2】

(図2)



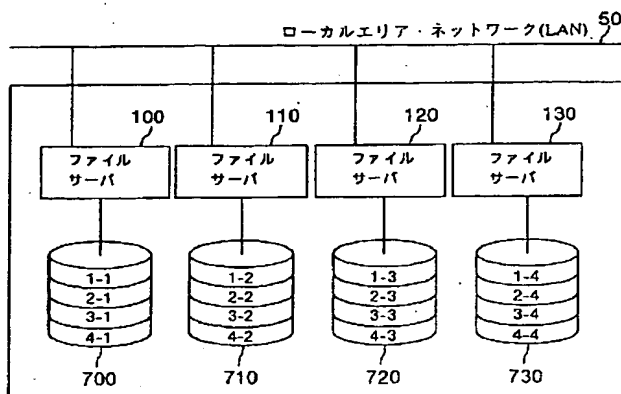
【図3】

(図3)



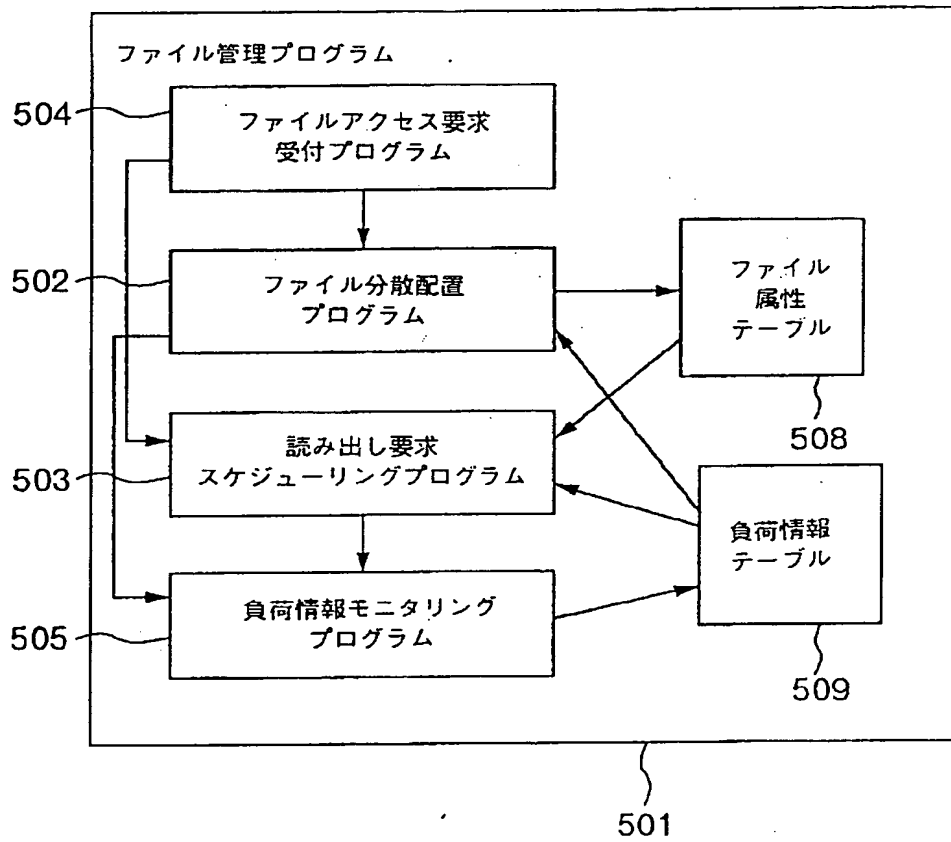
【図15】

図 15



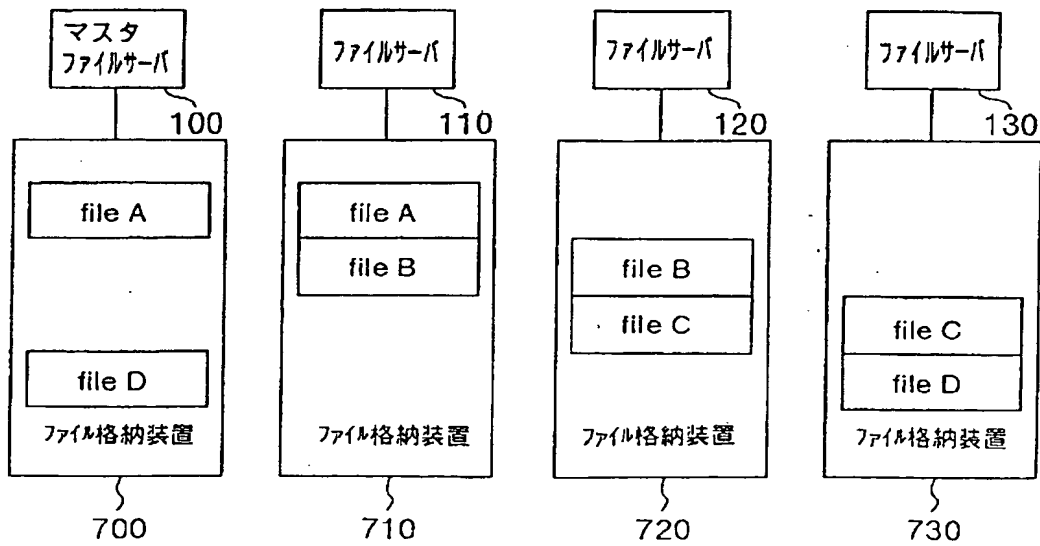
【図4】

図4



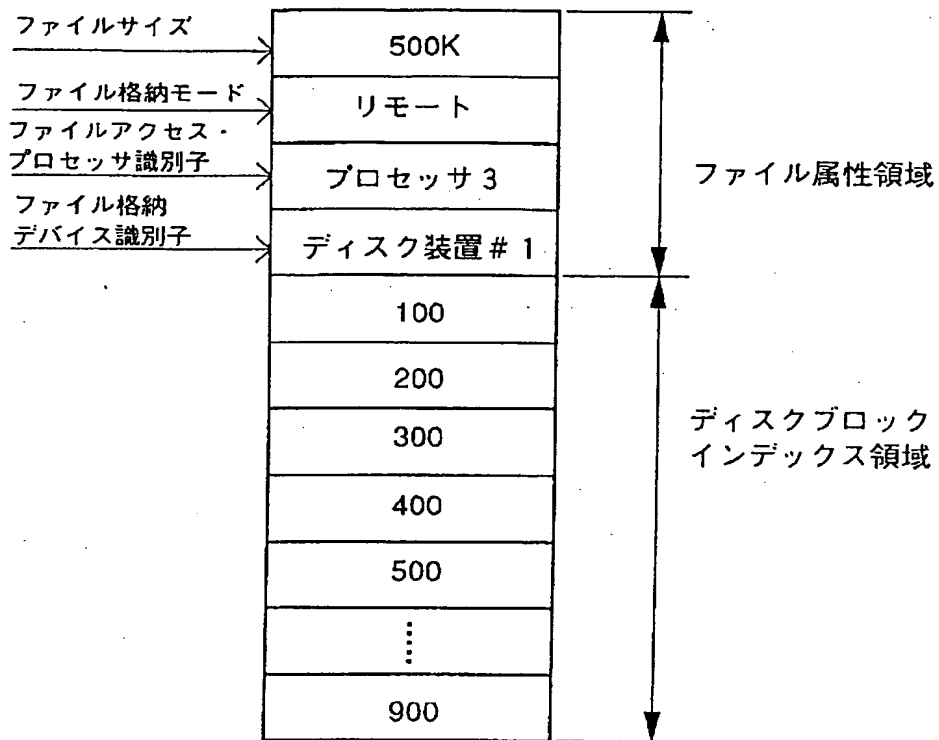
【図10】

図10



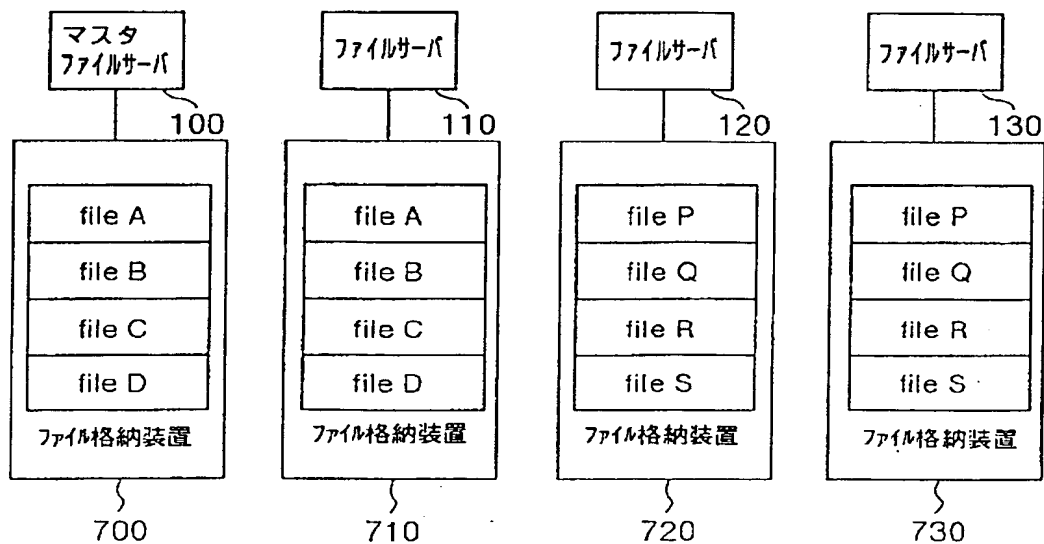
【図5】

図5



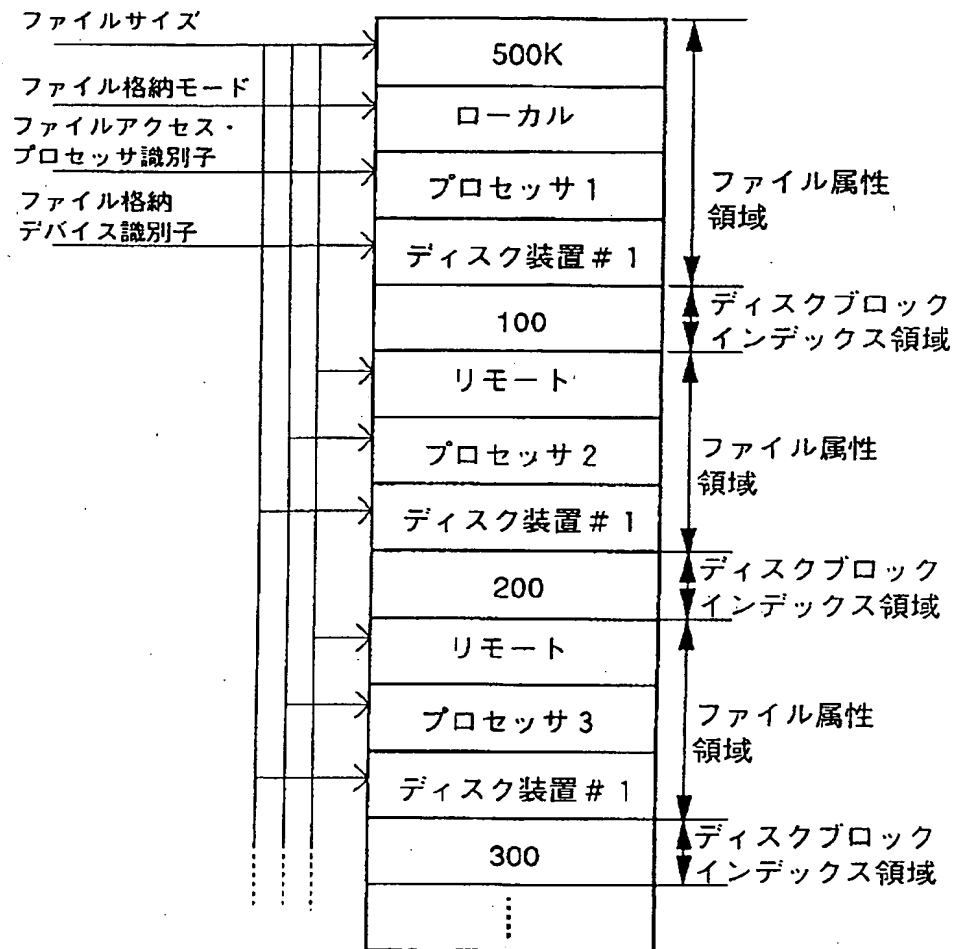
【図11】

図11



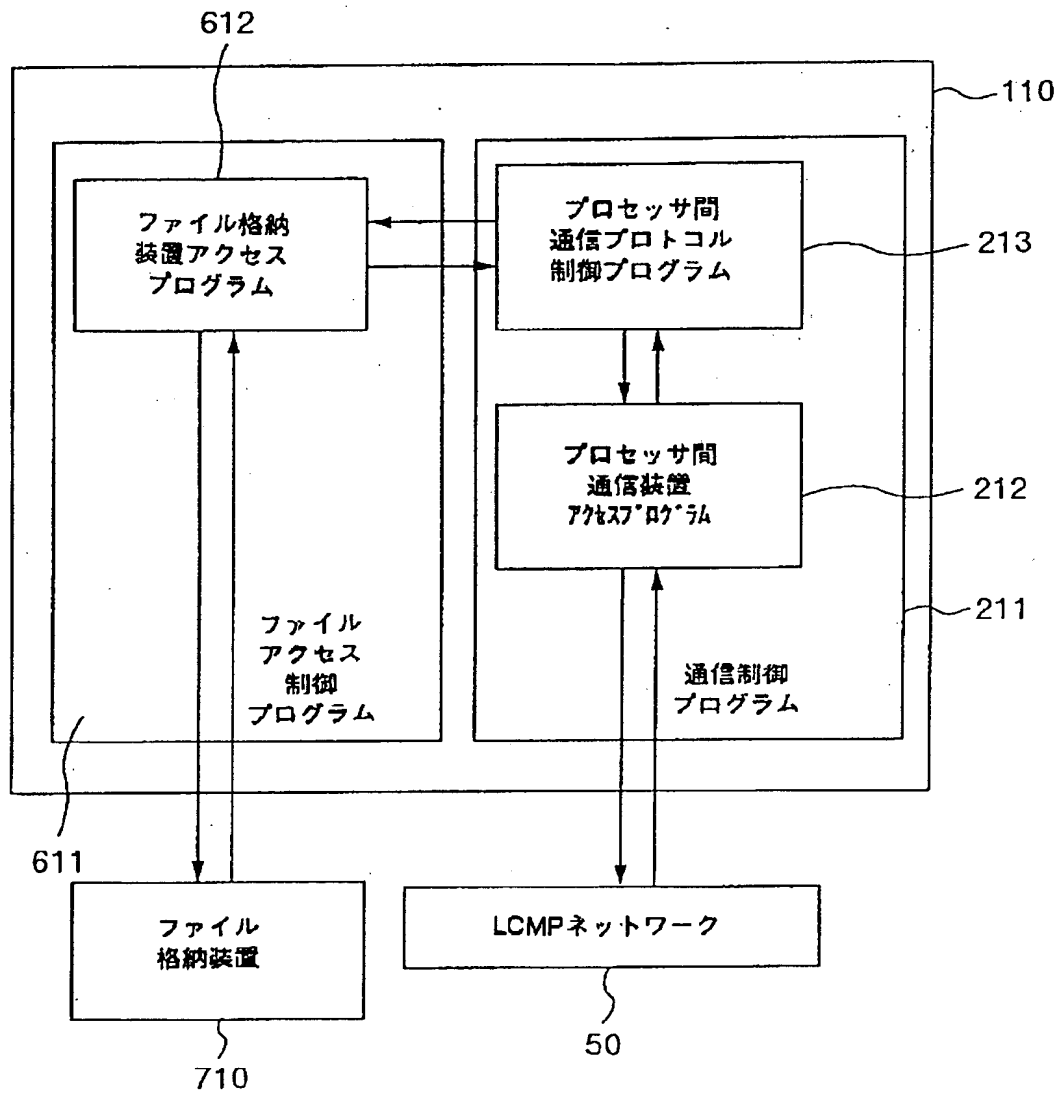
【図6】

図6



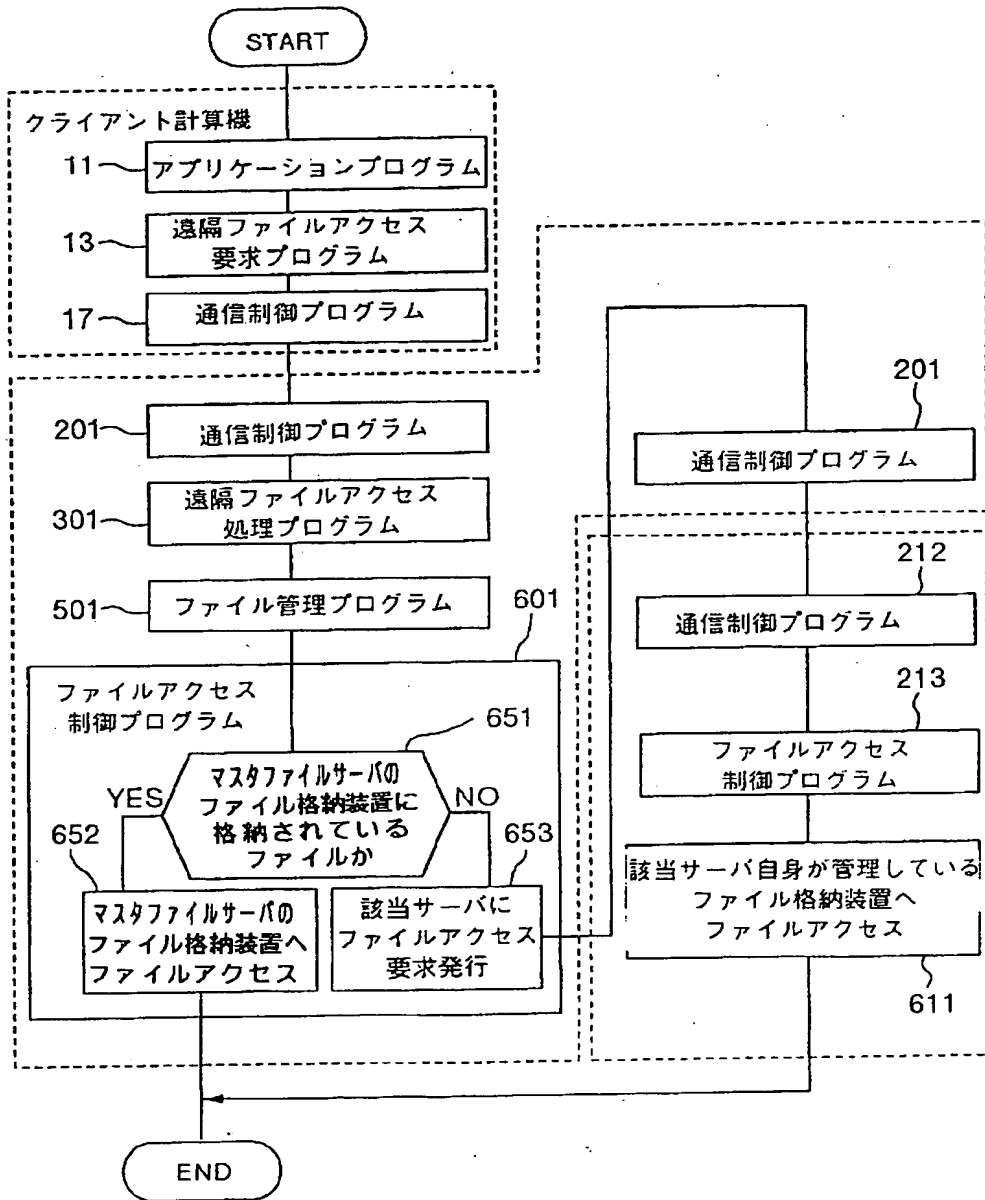
【図7】

図7



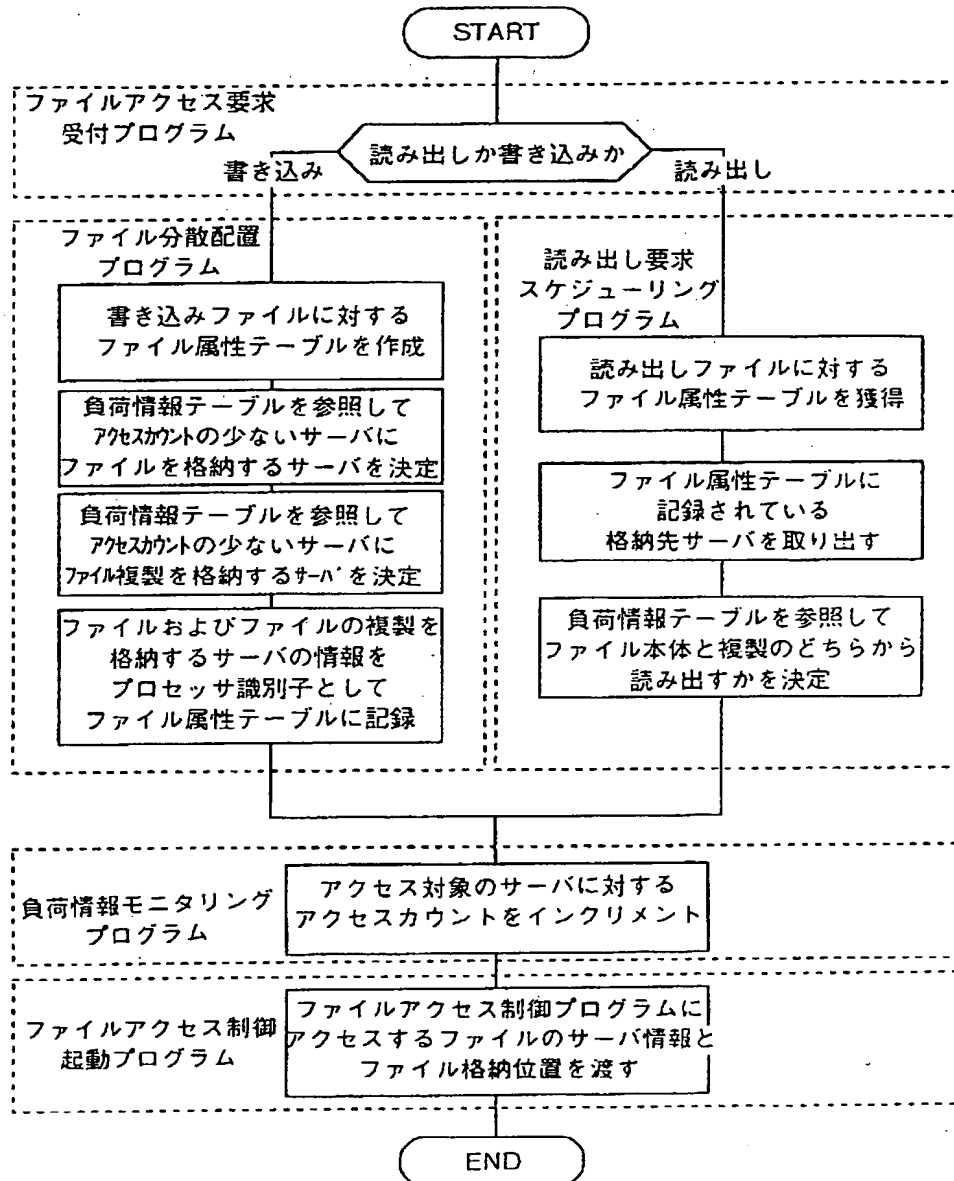
【図8】

図 8

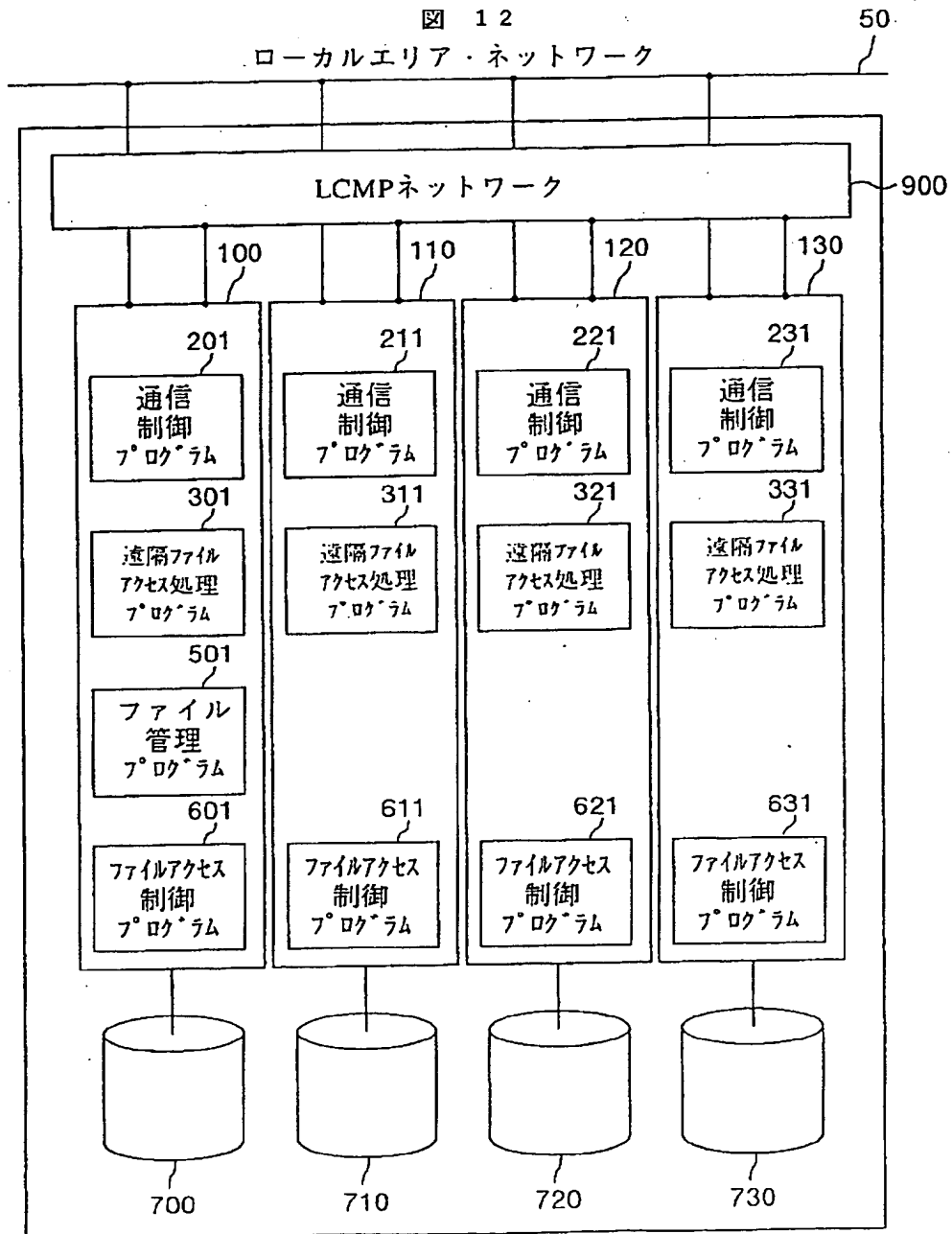


【図9】

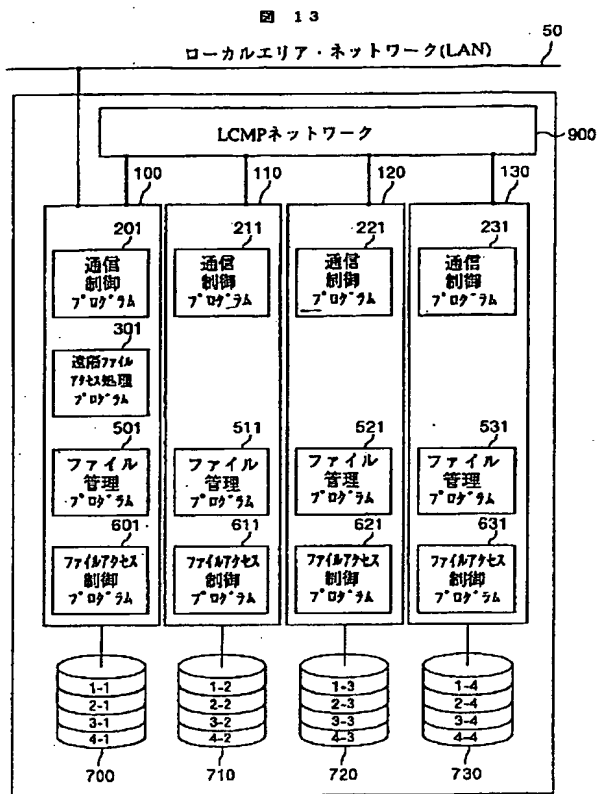
図 9



【図12】

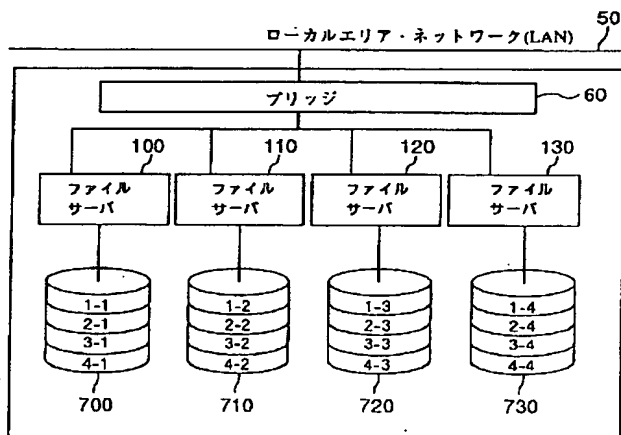


【図13】

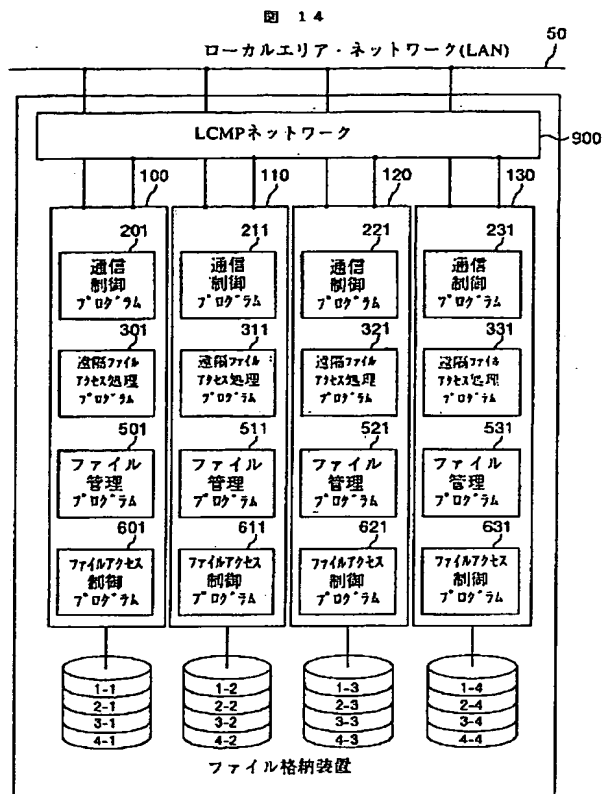


【図16】

図 16

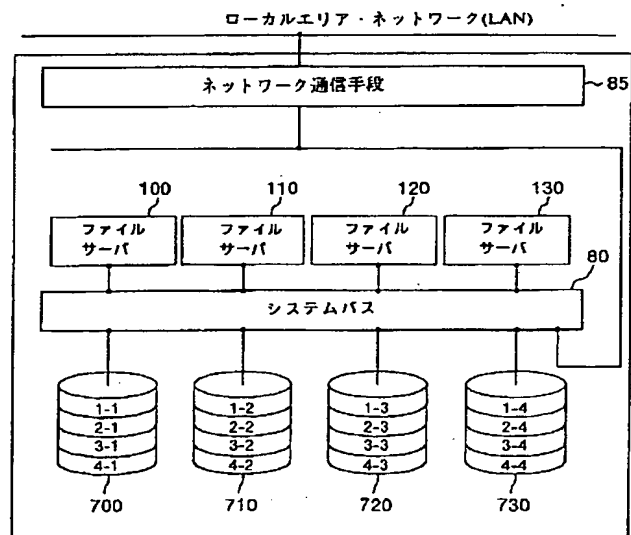


【図14】



【図17】

図 17



フロントページの続き

(72)発明者 山下 洋史
東京都国分寺市東恋ヶ窪1丁目280番地
株式会社日立製作所中央研究所内

(72)発明者 多田 勝己
東京都国分寺市東恋ヶ窪1丁目280番地
株式会社日立製作所中央研究所内

(72)発明者 川口 久光
東京都国分寺市東恋ヶ窪1丁目280番地
株式会社日立製作所中央研究所内

(72)発明者 加藤 寛次
東京都国分寺市東恋ヶ窪1丁目280番地
株式会社日立製作所中央研究所内

(72)発明者 鬼頭 昭
神奈川県横浜市戸塚区戸塚町5030番地 株
式会社日立製作所ソフトウェア開発本部内

(72)発明者 山田 秀則
神奈川県秦野市堀山下1番地 日立コンピ
ュータエンジニアリング株式会社内

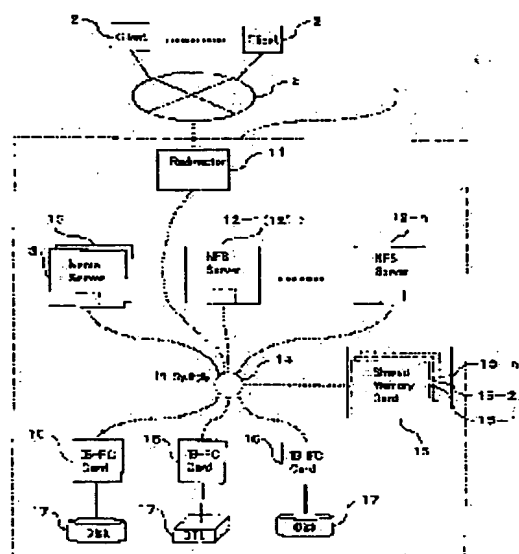
(11)Publication number : 2002-163140
(43)Date of publication of application : 07.06.2002

G06F 12/00
G06F 13/10

(71)Applicant : FUJITSU LTD

(72)Inventor : OE KAZUICHI
NISHIKAWA KATSUHIKO

SOLUTION: This storage system is provided with a storage device 17 capable of storing file data, a plurality of file servers 12-1 through 12-n performing file processes in response to requests on file data to the storage device 17, a file server management node 11 managing the transfer processes of the file requests received from clients 2 via an external network 3 to the file servers 12-i (i=1 through n) and the response processes to the clients 2 for the file requests, and the internal network 14 communicatably connecting the storage device 17, the file servers 12-i, and the file server management node 11 together.



[Date of extinction of right]

<http://www1.ipdl.jpo.go.jp/PA1/result/detail/main/wAAAa07213DA414163140P1.htm> 2003/05/21

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. *** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

- [Claim 1] The storage which can memorize file data, and two or more file servers which perform file processing according to the request about this file data to this storage. The transfer processing to this file server of the request received from a client through an external network. The file server management node which carries out unitary management of the response processing to this client to this request. The storage system characterized by having offered the internal network which interconnects this storage, this file server, and this file server management node possible [communication], and being constituted.
- [Claim 2] The storage system according to claim 1 characterized by connecting the name server which carries out unitary management of the file data name which this file server treats to this internal network.
- [Claim 3] The storage system according to claim 1 characterized by connecting the shared memory with accessible this file server management node and this file server to this internal network.
- [Claim 4] The storage system according to claim 2 characterized by connecting the shared memory with accessible this file server management node, this file server, and this name server to this internal network.
- [Claim 5] A storage system given in any 1 term of claims 1-4 characterized by having offered the request analysis section in which this file server management node analyzes the content of this request, and the request transfer section which transmits this request to a specific file server according to the analysis result of this request analysis section.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. *** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[The technical field to which invention belongs] this invention relates to the storage system which enables sharing of file data by two or more clients by connecting with a desired network about a storage system.

[0002]

[Description of the Prior Art] As conventional technique of realizing sharing (only henceforth "file sharing") of the file data between two or more nodes (client) on a network. For example, as typically shown in drawing 16, a file server 200 is built on the desired networks 100, such as LAN (Local Area Network), using a Network File System (NFS: Network File System). A secondary storage 400 is connected to this file server 200 through the interfaces 300, such as SCSI (Small Computer System Interface: -- generally called "SCSI"). The method of realizing file sharing between two or more clients 500 in this secondary storage 400 is learned well.

[0003] However, the following technical problems occur by this method.

** Special skill is required to build and maintain a file server (maintenance).

** Extension (capacity, access performance) of a file server is not easy. Even if extensible, maintenance cost will increase by a file server being divided into plurality etc.

[0004] ** Special skill is required for the system construction and maintenance (maintenance) offered at the time of failure, and the costs for it also start.

NAS (Network Attached Storage) is proposed as a method of solving these technical problems in recent years. It is the system which the portion (refer to the dashed line frame in drawing 16) which consists of an above-mentioned file server 200 and an above-mentioned secondary storage 400 is equivalent to what was beforehand built as one storage system, connects with a network 100, and this NAS only performs an easy setup, and can realize file sharing, and special skill is unnecessary to construction and maintenance of a system (maintenance).

[0005]

[Problem(s) to be Solved by the Invention] However, also in such NAS, the technical problem that the scalability which can fully respond to band expansion (it will be about 10Gbps after 1Gbps and several years) in the present condition) of LAN which is progressing quickly now by the low cost is not obtained remains. That is, when it is going to correspond to band expansion of the network of a connection place, also in NAS, simply, an internal file server and an internal secondary storage will be extended, consequently a file server will be divided into plurality, and the secondary storage which each manages will also be divided.

[0006] That is, an above-mentioned file server 200 and an above-mentioned secondary storage 400 will be arranged in parallel and (independent) prepared. For this reason, it will be necessary to maintain a file server separately (maintenance), and maintenance cost will increase after all. It was originated in view of such a technical problem, and this invention aims at offering a storage system with the scalability which can fully respond by the low cost to band expansion of a network.

[0007]

[Means for Solving the Problem] In order to attain the above-mentioned purpose, the storage

system (claim 1) of this invention. The storage which can memorize file data, and two or more file servers which perform file processing according to the request to this storage. The transfer processing to the above-mentioned file server of the request received from a client through an external network. It is characterized by having offered the internal network which interconnects the file server management node which carries out unitary management of the response processing to the client to the request, and storage, an above-mentioned file server and an above-mentioned file server management node possible [communication], and being constituted.

[0008] Here, the name server which carries out unitary management of the file data name which the above-mentioned file server treats may be connected to the above-mentioned internal network, and the shared memory with accessible above-mentioned file server management node and above-mentioned file server may be connected to it (claim 3). (claim 2) In addition, when the above-mentioned name server exists, in addition to an above-mentioned file server management node and an above-mentioned file server, this name server can also be accessed at the above-mentioned shared memory (claim 4).

[0009] Moreover, it is desirable for the above-mentioned file server management node to offer the request analysis section which analyzes the content of the above-mentioned request, and the request transfer section which transmits this request to a specific file server according to the analysis result of this request analysis section, and to be constituted (claim 5).

[0010]

[Embodiments of the Invention] Hereafter, the gestalt of operation of this invention is explained with reference to a drawing.

(A) The explanatory drawing 1 of 1 operation gestalt is a block diagram showing the storage structure of a system (storage architecture) as 1 operation gestalt of this invention. The storage system 1 (only henceforth "a system 1") shown in this drawing 1 is for realizing file sharing among two or more clients 2 connected to the external network (for example, Gigabit Ethernet (registered trademark)). 3. A redirector (Redirector) 11, two or more NFS servers (file server) 12-1 ~ 12-n, a name server 13, a shared memory (Shared Memory) 15, the IB-FC card 16, and a secondary storage 17 [a disk unit, a tape unit (DTL), etc.]. It has offered. It has each of these components 11, 12-i, and the composition that 13, 15, 16, and 17 were mutually connected through the high-speed (interior) network [in FINI band (Infiniband)] switch 14 which is a 4-10Gbps (gigabit/second) grade.

[0011] Thus, if NFS server 12-i and a secondary storage 17 are connected to the internal network 14 if needed by taking the gestalt (clustering) which connects each components inside a system (a redirector 11, NFS server 12-i, a name server 13, secondary storage 17, etc.) in the internal network 14, it can extend easily, and the expandability (capacity, access performance, etc.) according to the transmission speed of the external network 3 etc. improves sharply.

[0012] Here, the above-mentioned redirector (file server management node) 11 is for carrying out unitary (concentration) management of the transfer processing to NFS server 12-i (=1-n) of the various request messages (only henceforth a "request") received from the arbitrary clients 2 through the external network 3, and the response processing to the client 2 of the request origin to the request. That is, there is no need of maintaining them separately like before by existence of this redirector 11 though NFS server 12-i and a secondary storage 17 are extended like ****.

[0013] In addition, the above-mentioned "request" means the demand about the file data (only henceforth a "file") memorized by the secondary storage 17, for example, has the file manipulation demands (writing / updating / read-out) (file access request) to the substance of file data, meta-information access of other refer to the file name, etc. Moreover, from each client 2, only IP (Internet Protocol) address given to this redirector 11 can be referred to fundamentally. That is, from each client 2, this system 1 appears as one file server.

[0014] That a function which was mentioned above should be realized and to this redirector 11. For example, Gigabit Ethernet card 11a which equips the interface for external network 3 as shown in drawing 2. In FINI band card 11b which equips the interface for the interior of a system (internal network), Network processor 11c for carrying out centralized control of the operation of

selves (redirector 11) including each of these cards 11a and 11b, Memory (primary storage) 11d for memorizing required software (program) and various data, when this network processor 11c operates etc. is mounted.

[0015] In addition, network processor 11c is connected with these components 11a-11d possible [two way communication] through PCI (Peripheral Component Interconnect) bus 11e. The above-mentioned network processor 11c here Transmission and reception (protocol conversion etc. is included) of the reply (response) message to the request sent and received between the internal network 14 and the external network 3, or its request etc., The analysis of the request (protocol) received from a client 2, the determination of the access file name based on the analysis result, the decision of destination NFS server 12-i of a receive request, etc. can be made, and the following control is also attained with this operation gestalt.

[0016] Namely, the request from a client 2 is analyzed, and it can be controlled, or it can be controlled [the request about the same file can be assigned to the same NFS server 12-i, and] now so that file access competition does not occur between the NFS servers 12-i so that a request is uniformly transmitted to each file server 12-i (to for example, NFS server 12-i with a light load).

[0017] For this reason, if its attention is paid to the function of the important section, the following function parts are mounted in this network processor 11c.

(1) As the request analysis section 111 which analyzes the content of the request from a client 2 ***** (2) Transfer history of the request of the past by the function (3) request transfer section 112 as the request transfer section 112 which transmits a receive request to specific NFS server 12-i according to the analysis result of this request analysis section 111 (for example) By performing in the background the function (4) NFS server load surveillance demon (daemon) 115 as the transfer history Records Department 113 which records at memory 11d the function as the load Monitoring Department 114 which supervises periodically the loaded condition of each NFS server 12-i --- by this, the above-mentioned request transfer section 112 Based on the transfer history by the above-mentioned transfer history Records Department 113, transmit the request to the file of the same file name to same NFS server 12-i, or it becomes possible to transmit a receive request to low NFS12-i of a load based on the load surveillance result by the NFS server load surveillance demon 115 (load Monitoring Department 114).

[0018] Therefore, since a request can be uniformly transmitted to each NFS server 12-i while processing speed improves sharply, what a load concentrates on a part of NFS server 12-i, and causes an obstacle can be prevented certainly, and reliability improves it sharply. In addition, this network processor 11c carries out the cache of the reply message for example, to meta-information access to cache memory 11f (memory 11d is sufficient) of the interior (refer to drawing 3). When the request about meta-information access is received from a client 2, it is cache memory 11f () first, or if memory 11d is checked and hit, a reply message is created, and it will return to a client 2 side as it is (refer to drawing 4) --- (without transmitting to NFS server 12-i or a name server 13) it can also be made like

[0019] This structure is applicable not only about meta-information but file data. However, it is better to select file data with high access frequency by network processor 11c, and to be made to carry out the cache only of the file data to cache memory 11f (or memory 11d), since required memory space increased when it was made to carry out the cache of all the file data in this case.

[0020] That is, the above-mentioned cache memory 11f () Or the function as the cache section which carries out the cache of the reply message to the client 2 about specific file data with high request frequency is also achieved memory 11d, Network processor 11c in this case is it cache memory 11f () that it is a thing about the file with the same new request. Or it will also have the function as the response section 116 (refer to drawing 2) which returns the reply message which carried out the cache to memory 11d to a client 2.

[0021] Thus, if a response is returned by the redirector 11 about the information (meta-information, file data) that access frequency is high, without carrying out a cache by the redirector 11 side, and transmitting a request to NFS server 12-i, the speed of response to the client 2 to the information that access frequency is high will improve sharply, and the processing speed and the throughput as this system 1 will improve by leaps and bounds.

[0022] Next, the above-mentioned NFS server 12-i can access file processing (writing / updating / read-out) according to the request transmitted from the redirector 11 at internal network (internal network switch) 14 course at a secondary storage 17, can be carried out, or can generate the reply message for sending the file-processing result as a response to the client 2 of a requesting agency, and can be transmitted to a redirector 11, respectively.

[0023] Each NFS server 12-i in addition, in hardware For example, as shown in drawing 5 Offer interface card (IB-IF) 12c which equips interfaces (protocol conversion etc.) with CPU(Central Processing Unit) 12a, memory (primary storage) 12b, and the internal network 14, and it is constituted. By CPU12a's reading the NFS server software (program) memorized by memory 12b, and operating, the function as NFS server 12-i mentioned above is realized.

[0024] Now, un-arranging [of being as becoming the same management file name by NFS server 12-i which is different conversely in spite of being different file data substance *****, and] may arise, and file access competition may arise between the NFS servers 12-i, / management file names differing by NFS server 12-i from which it differs here although such NFS server 12-i is the same file data substance when a file name is managed uniquely, respectively

[0025] The above-mentioned name server 13 solves such un-arranging. That is, in this name server 13, by carrying out unitary management of the meta-information access from NFS server 12-i, management file name space in all NFS server 12-i is set to one, and the file access competition between the NFS servers 12-i is avoided. Therefore, the reliability of file sharing by this system 1 will improve sharply by offering this name server 13.

[0026] In addition, it is shown in drawing 1 --- as --- this name server 13 --- heterologies (down), such as failure, --- offering --- present --- business and the object for reserves exist moreover, in hardware also about these name servers 13 Interface card 13c which equips an interface with the same composition as NFS server 12-i (refer to drawing 5), i.e., CPU13a, memory (primary storage) 13b, and the internal network 14 is mounted. The function as a name server 13 mentioned above is realized by CPU13a's reading the name server software (program) memorized by memory 13b also in this case, and operating.

[0027] Next, the redirector 11 of the above [the above-mentioned shared memory 15], NFS server 12-i, and a name server 13 are accessible memory through the internal network 14, respectively, for example, certain NFS server 12-i --- or --- present, when the name server 13 of business is downed (at the time of obstacle generating) the NFS server 12-i --- or --- present --- the information for taking over processing of the name server 13 of business to the name server 13 other NFS server 12-k (it being k=i at k= 1 - n), or for reserves --- (Following and taking over information) etc. --- server 12-i and 13 --- it is held independently at memory card (Shared Memory Card) 15-1 - 15-m (m is the natural number) (refer to drawing 6 and drawing 8) (backup)

[0028] That is, each above-mentioned NFS server 12-i (CPU12a) or the name server 13 (CPU13a) will have the function as the taking over information Records Department 121 (131) (refer to drawing 5) which succeeds information required to offer at the time of a heterology and take over processing to the name server 13 other NFS server 12-i or for reserves, and is recorded on a shared memory 15 as information.

[0029] In addition, the down of NFS server 12-i is typically shown in drawing 6 --- as --- present --- the name server 13 (CPU13a) of business performs the NFS server surveillance demon 132 in the background --- supervising --- present --- as typically shown in drawing 8 , the down of the name server 13 of business is supervised because the name server 13 (CPU13a) for reserves performs the name server surveillance demon 133 in the background

[0030] And as typically shown in drawing 7 , when the down of NFS server 12-i is detected (Step S1) As opposed to NFS server 12-k (for example, NFS server 12-k with a light load) other than NFS server 12-i to which the name server 13 (CPU13a) of business was downed present --- While directing to succeed processing of downed NFS server 12-i, the down of NFS server 12-i is notified to a redirector 11 (Step S2).

[0031] NFS server 12-k (CPU12a) which received taking over directions accesses a shared memory 15 through the internal network 14 by this, and processing of NFS server 12-i which read the taking over information backed up there and was downed is succeeded (Step S3). On

the other hand, a redirector 11 (network processor 11c) is made not to transmit the request by the request transfer section 112 by receiving the above-mentioned notice from a name server 13 to down NFS server 12-i at this time.

[0032] That is, the name server 13 (CPU13a) in this case The malfunction detection section 134 (refer to drawing 6) which detects the heterology of NFS server 12-i, if the heterology of NFS server 12-i is detected in this malfunction detection section 134 It will have the function as the taking over directions section 135 (refer to drawing 6) which gives [succeeding the processing of NFS server 12-i which carried out the heterology to other NFS server 12-k other than the NFS server 12-i based on the taking over information on a shared memory 15, and] taking over directions.

[0033] Thus, with this operation gestalt, since the name server 13 other NFS server 12-k and for reserves can succeed processing even if NFS server 12-i and a name server 13 are downed, file processing normal as a storage system 1 can be continued, and obstacle-proof nature improves sharply. In addition, although this example is explanation about redundancy-izing of NFS server 12-i or a name server 13, of course, it is also possible to redundancy-ize a redirector 11 similarly, moreover --- present --- you may make it take over to either of the NFS server 12-i depending on the case about the taking over at the time of the name server 13 down of business

[0034] Next, the redirector 11 at the time of the request transfer to NFS server 12-i from the redirector 11 mentioned above and the concrete processing by NFS server 12-i are explained. A redirector 11 will analyze the file access request in the request analysis section 111, if the file access request for writing in a certain file data from a client 2 is received.

[0035] In addition, as shown in drawing 12, the above-mentioned "file access request" has the header unit 21 which consists of physical-layer header (Phy Header) 21a, IP header (Internet Protocol Header) 21b, TCP header (Transmission Control Protocol Header) 21c, and NFS header 21d etc., and the real file data section 22 in which the substance (real file data) of file data which should actually be written in a secondary storage 17 was stored, and changes.

[0036] And the request analysis section 111 asks for the boundary of the position where the real file data under above-mentioned file access request starts, i.e., a header unit 21 and the real file data section 22, as header offset value [number-of-bits ("a") etc. 23 from boundary information; for example, a head], as typically shown in drawing 9. The boundary information 23 searched for is notified to the request transfer section 112, and the request transfer section 112 appends the boundary information 23 notified to the above-mentioned file access request (addition), and sends it to NFS server 12-i of the destination.

[0037] That is, as shown in drawing 2, the above-mentioned request analysis section 111 The function as header offset value analysis section 111a to calculate the header offset value 23 which analyzes the received file access request, and expresses the boundary position of the header unit 21 of the file access request, and the real file data section 22. It will have the function as header offset pricing Kabe 111b which adds the header offset value 23 acquired by this header offset value analysis section 111a to the file access request to which it is transmitted to NFS server 12-i.

[0038] Then, in the NFS server 12-i side, it is based on the header offset value 23 added by the redirector 11 side like ****. The NIC (Network Interface Card) driver (network driver) 122 The real file data section 22 and the other field The start address of (a header unit 21) can be assigned to the page boundary [the page boundary (another field); buffer (mbuf) 123,124] of the message treated within the kernel of a high order layer (NFS processing layer) (kernel high order layer), respectively (refer to drawing 10).

[0039] When are done in this way and a file access request reaches the file system section 125 of a kernel high order layer as typically shown, for example in drawing 11, it becomes possible to move data to the file system buffer 126 only by changing the start address (pointer) of the real file data section 22 for the pointer to the file system buffer 126, without generating a copy of data (map change) (the zero copy in a kernel is realized). Therefore, DMA (Direct Memory Access) can also be performed at high speed, and can improve sharply the processing speed and the throughput of NFS server 12-i.

[0040] In addition, although making it ask by the NIC driver 122 side is also thought of, the boundary of a header unit 21 and the real file data section 22 In this case, since the amount of processings (header analysis) of the NIC driver 122 increases (the NIC driver 122 usually performs only analysis of physical-layer header 21a). The direction which asked for the boundary by the redirector 11 side which has the analysis feature (request analysis section 111) of a header unit 21 from the first as mentioned above The kernel zero copy in a high order layer (NFS processing layer) can be realized, without increasing the throughput in a NIC driver layer (** which does not cause a throughput fall).

[0041] According to the storage system 1 of this operation gestalt, as mentioned above, to the system 1 interior By having formed a redirector 11, two or more NFS server 12-i, the name server 13, the shared memory 15, and the secondary storage 17, and having made these the composition connected in the high-speed internal network 14 Since NFS server 12-i and a secondary storage 17 can be extended easily if needed and it moreover is not necessary to maintain NFS server 12-i separately (maintenance) The performance (for example, it can respond to 10GbpsLAN) and capacity scalability which can fully respond by the low cost to band expansion of the external network 3 are securable.

[0042] Especially with the operation gestalt mentioned above, a redirector 11 so that processing may be uniformly assigned to each NFS server 12-i according to the load of each NFS server 12-i control or Assign the processing to the request about the same file to the same NFS12-i, or Since it answers by the reply message which carried out the cache by the redirector [not NFS server 12-i but] 11 side to the request about the high file of access frequency The processing speed and performance are improving by leaps and bounds, and the performance and the capacity scalability which can certainly respond are realized to 10GbpsLAN.

[0043] (B) NFS server 12-i with a larger capacity of memory 12b than other NFS server 12-i is arranged as cache server 12' (refer to drawing 1.), and you may make it the file access in this cache server 12' return a response only by the R/W to memory 12b fundamentally in the system 1 in which the 1st modification carried out explanation **** at a client 2 side.

[0044] And about the high file of the access frequency whose request frequency within a fixed period is more than the number of times of predetermined (threshold) the cache is carried out to memory 12b of cache server 12', and it is made for cache server 12' to answer. Specifically, first, the access frequency of each file is supervised by the redirector 11 side, and about access to a file with access frequency higher than a certain threshold, directions are given to a name server 13, NFS server 12-i, and cache server 12' from redirector 12-i so that it may process by cache server 12-i.

[0045] That is, at this time, a redirector 11 (request transfer section 112) will transmit the request about the high file of access frequency to cache server 12'. Thereby, since access to the high file of access frequency is processed within cache server 12', without accessing a secondary storage 17, it contributes to the large improvement in the processing speed as a storage system 1, and a throughput greatly.

[0046] On the other hand, if the access frequency to the high file of the above-mentioned access frequency falls, a redirector 11 directs to assign suitable (for example, a load --- light) NFS server 12-i, and to shift processing to a name server 13, NFS server 12-i, and cache server 12' (if the access frequency to the file by which the cache is carried out to memory 12b of cache server 12' becomes below the number of times of predetermined)

[0047] That is, a redirector 11 (request transfer section 112) changes the destination of a request into NFS server 12-i other than cache server 12' in this case. While it is avoided that the cache of the file from which access frequency has fallen continues being forever carried out to cache server 12' by this, consequently it can cut down memory space required for cache server 12', it can give a margin to processing by cache server 12', and can improve the throughput.

[0048] (C) It is also possible to enable it to access a secondary storage 17 also from the external node 19 by [which is explanation of the 2nd modification] connecting the above-mentioned secondary storage 17 with a name server 13 and NFS server 12-i by FC switch 18 course (a secondary-storage network being built), and connecting the FC switch 18 and the

external node 19, as shown in drawing 13. However, the file system operated by the external node 19 in this case needs to be the same as the file system in the storage system 1.

[0049] Although a certain mediation control is needed about access from the external node 19 by doing in this way in order to avoid the access competition with NFS server 12-i of the system 1 interior, access to the file in this storage system 1 from the external node 19 is attained. Since the management file name in a system 1 will be followed also about access from the external node 19 if access to a name server 13 from the external node 19 is permitted as it corrects, for example, is shown in drawing 14, the file access from the external node 19 can be performed without needing the above-mentioned mediation control. In addition, although it is the case where access to the name server 13 in internal network 14 course is permitted, in this drawing 14, you may make it permit access by secondary-storage network (FC switch 18) course in drawing 13, of course.

[0050] Moreover, as shown, for example in drawing 15, you may communalize NFS server 12-i and the external node 19. That is, it constitutes so that file processing according to the request directly received from the external network 3 in NFS server 12-i may be performed to a secondary storage 17. When NFS server 12-i functions as a file server of the storage system 1 mentioned above by this when a certain client 2 has accessed by the redirector 11 course, and NFS server 12-i has been accessed directly, it will function as a usual file server which answers without going via a redirector 11. That is, the both sides of access by the NFS server 12-i course from a client 2 and the direct access which does not go via NFS server 12-i are permitted.

[0051] The direct access from the outside of any case becomes possible, and fusion to other storage architecture [SAN (Storage Area Network) etc.] can be realized. In addition, in drawing 14 and drawing 15, a sign 20 expresses the network disk adapter which equips the interface of the internal network 14 and a secondary storage 17.

[0052] Moreover, although the shared memory 15 mentioned above is omitted with the composition shown in drawing 13 - drawing 15, of course, it may be equipped. If it does in this way, also in the composition shown in drawing 13 - drawing 15, the same backup processing as the above will be attained.

(D) In addition, although the operation gestalt mentioned above, in addition, explained the case where Gigabit Ethernet was applied as Infiniband and an external network 3 as an internal network 14, of course, it is also possible to carry out a system construction using other high-speed networks other than these.

[0053] Moreover, it is not necessary to necessarily offer an above-mentioned name server 13 and an above-mentioned shared memory 15, and even if it omits these either or both sides, the purpose of this invention is attained enough. Furthermore, although NFS is applied to the file server with the operation gestalt mentioned above, this invention is not limited to this but, of course, it is also possible to apply other file systems.

[0054] Moreover, although premised on the capacity (transmission speed) of the internal network 14 being about 4-10Gbps with the operation gestalt mentioned above, this speed can respond, if it changes suitably according to band expansion of the external network 3. And this invention is not limited to the operation gestalt mentioned above, but in the range which does not deviate from the meaning of this invention besides the above, can deform variously and can be carried out.

[0055] (E) Additional remark 1] Storage which can memorize file data, Two or more file servers which perform file processing according to the request to this storage. The transfer processing to this file server of the request received from a client through an external network. The file server management node which carries out unitary management of the response processing to this client to this request. The storage system characterized by having offered the internal network which interconnects this storage, this file server, and this file server management node possible [communication], and being constituted.

[0056] [Additional remark 2] Storage system of the additional remark 1 publication characterized by connecting the name server which carries out unitary management of the file data name which this file server treats to this internal network.

[Additional remark 3] Storage system of the additional remark 1 publication characterized by connecting the shared memory with accessible this file server management node and this file server to this internal network.

[0057] [Additional remark 4] Storage system of the additional remark 2 publication characterized by connecting the shared memory with accessible this file server management node, this file server, and this name server to this internal network.

[Additional remark 5] Storage system given in any 1 term of additional remarks 1-4 characterized by having offered the request analysis section in which this file server management node analyzes the content of this request, and the request transfer section which transmits this request to a specific file server according to the analysis result of this request analysis section.

[0058] [Additional remark 6] Storage system of the additional remark 5 publication which offers the transfer history Records Department where this file server management node records the transfer history of the request of the past by this request transfer section, and is characterized by being constituted so that this request transfer section may transmit the request to the file data of the same file data name to the same file server based on this transfer history of this transfer history Records Department.

[0059] [Additional remark 7] Storage system of the additional remark 5 publication which this file server management node offers the load Monitoring Department which supervises the load of this file server, and is characterized by constituting this request transfer section so that this request may be transmitted to the low file server of a load based on the surveillance result in this load Monitoring Department.

[0060] [Additional remark 8] Storage system of the additional remark 5 publication which offers the primary storage to which at least one in this file server can carry out the cache of the file data in this storage, and is characterized by being constituted as a cache server which performs file processing according to this request in this primary storage.

[Additional remark 9] Storage system of the additional remark 8 publication characterized by being constituted so that this request transfer section may transmit the request about file data with the above-mentioned high request frequency to this cache server while this primary storage of this cache server was constituted so that the request frequency within a fixed period might carry out the cache of the file data more than the number of times of predetermined.

[0061] [Additional remark 10] Storage system of the additional remark 9 publication characterized by being constituted so that the destination of this request may be changed into file servers other than this cache server when the request frequency to this file data by which the cache of this request transfer section is carried out to this primary storage of this cache server became below the number of times of predetermined.

[0062] [Additional remark 11] The storage system of the additional remark 5 publication which carries out [having offered the header offset value analysis section which calculates the header offset value as which this request analysis section analyzes this request, and expresses the boundary position of the header unit of the request concerned, and the real file data section, and header offset pricing Kabe who add this header offset value acquired in this header offset value analysis section to this request to which it is transmitted to this file server, and] as the feature.

[0063] [Additional remark 12] Storage system of the additional remark 11 publication

characterized by having offered the network driver by which this file server copies this header unit and these data division of this request to the field to which the messages treated in a kernel high order layer, respectively differ based on this header offset value added to this request.

[Additional remark 13] Storage system given in any 1 term of additional remarks 1-12

characterized by having offered the cache section to which this file server management node carries out the cache of the response message to this client about specific file data with high request frequency, and the response section which returns the applicable response message of this cache section to this client as this request is a thing about this specific file data.

[0064] [Additional remark 14] Storage system given in the additional remark 3 or additional remark 4 characterized by having offered the taking over information Records Department which

succeeds information required for this file server to offer at the time of a heterology, and take over processing to other file servers, and records on this shared memory as information.

[Additional remark 15] Storage system of the additional remark 14 publication characterized by preparing the malfunction detection section which detects the heterology of this file server, and the taking over directions section which will give [succeeding processing of this unusual file server based on this taking over information on this shared memory to other file servers other than the file server (henceforth an unusual file server) concerned and] taking over directions if the heterology of this file server is detected in this malfunction detection section.

[0065] [Additional remark 16] Storage system given in any 1 term of additional remarks 1-15 characterized by constituting this storage so that access from an external node may be permitted.

[Additional remark 17] Storage system given in any 1 term of additional remarks 2-15 characterized by constituting this name server so that access from an external node may be permitted.

[0066] [Additional remark 18] Storage system given in any 1 term of additional remarks 1-17 characterized by being constituted so that this file server may perform file processing according to the request received directly from this external network to this storage.

[0067]

[Effect of the Invention] Two or more file servers which perform file processing according to the request to storage according to the storage system of this invention as explained in full detail above. Since the internal network which interconnects the file server management node which carries out unitary management of the processing of each file server, and storage, a file server and a file server management node possible [communication] is offered Since a file server and storage can be extended easily if needed and it moreover is not necessary to maintain each file server separately (maintenance) The performance and capacity scalability which can fully respond by the low cost to band expansion of an external network are securable.

[0068] And if the name server which carries out unitary management of the file data name which each above-mentioned file server treats to the above-mentioned internal network is connected, since it can prevent that file access competition arises between file servers, it contributes to the improvement in reliability of file sharing greatly. Moreover, though an obstacle occurs in some file servers, since processing normal as a storage system is continuable by memorizing the taking over information on the file server which connected the shared memory to the above-mentioned internal network, and was offered on it at this shared memory at the time of obstacle generating at any time, obstacle-proof nature improves sharply.

[0069] Furthermore, if a file server management node may be constituted so that the request to the file data of the same file data name may be transmitted to the same file server based on the transfer history of the past request, and it does in this way, its processing speed will improve sharply. Moreover, if the load of a file server is supervised, you may make it transmit a request to the low file server of a load and it does in this way, since a request can be uniformly transmitted to each file server, what a load concentrates on some file servers and causes an obstacle can be prevented certainly, and reliability improves it sharply.

[0070] Furthermore, about file data with high request frequency, if the cache is carried out to the primary storage of a cache server and it is made to process by the cache server, since the access frequency to storage is sharply reducible, processing speed and processability ability improve further. And if the request frequency to the file data by which the cache is carried out to the primary storage of a cache server in this case becomes below the number of times of predetermined and it will be made to process by file servers other than a cache server Since it is avoided that the low file data of request frequency continues being held forever at a cache server While memory space required for the primary storage of a cache server is reducible, a margin can be given to processing by the cache server and the throughput can be improved.

[0071] Moreover, in a file server management node, the header offset value showing the boundary position of the header unit of a request and the real file data section is calculated, and if the header offset value is added to a request and it is made to transmit to a file server, in the network driver of a file server, the header unit and data division of a request can be copied to

the field to which the messages treated in a kernel high order layer, respectively differ based on the header offset value. Therefore, the zero copy in a kernel can be realized and the processing speed and the throughput of a file server can be improved sharply.

[0072] Furthermore, if the response message which carries out the cache of the response message to the client about specific file data with high request frequency, and carried out the cache to the received request being a thing about the file data in the above-mentioned file server management node is returned to a client, since there will be no need of transmitting a request to a file server, the speed of response to a client will improve sharply, and the processing speed and the throughput as this system will improve by leaps and bounds.

[0073] Moreover, if the above-mentioned storage may be constituted so that access from an external node may be permitted, and it does in this way, fusion to other storage architecture will be attained. If access to the above-mentioned name server [node / external] is permitted here, the file access from an external node will become possible, without needing file access mediation control with the above-mentioned file server and an external node.

[0074] Furthermore, if the above-mentioned file server may be constituted so that file processing according to the request received directly may be performed from the above-mentioned external network to storage, and it does in this way, since the both sides of access via the file server from a client and the direct access which does not go via a file server are permissible, fusion to other storage architecture is attained also in this case.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

- [Brief Description of the Drawings]
- [Drawing 1] It is the block diagram showing the storage structure of a system (storage architecture) as 1 operation gestalt of this invention.
- [Drawing 2] It is the block diagram showing the composition of a redirector shown in drawing 1.
- [Drawing 3] It is a block diagram for explaining the case where the cache of the meta-information is carried out by the redirector shown in drawing 1.
- [Drawing 4] It is a block diagram for explaining the case where a reply message is returned by the redirector shown in drawing 1.
- [Drawing 5] It is the block diagram showing the composition of an NFS server (name server) shown in drawing 1.
- [Drawing 6] It is a block diagram for explaining the case where the taking over information on an NFS server is backed up to the shared memory shown in drawing 1.
- [Drawing 7] It is a block diagram for explaining the case where processing of the NFS server which was backed up by the shared memory shown in drawing 6 and which succeeded and was downed based on information is succeeded.
- [Drawing 8] It is a block diagram for explaining the case where the taking over information on a name server is backed up to the shared memory shown in drawing 1.
- [Drawing 9] It is a block diagram for explaining the case where the boundary information on a file access request is searched for in the redirector shown in drawing 1.
- [Drawing 10] It is a block diagram for explaining the case where the zero copy in a kernel is realized based on the boundary information on the file access request shown in drawing 9 in the NFS server shown in drawing 1.
- [Drawing 11] It is a block diagram for explaining the case where the zero copy in a kernel is realized based on the boundary information on the file access request shown in drawing 9 in the NFS server shown in drawing 1.
- [Drawing 12] It is drawing showing the example of a format of the file access request shown in drawing 9 - drawing 11.
- [Drawing 13] It is the block diagram showing the composition in the case of permitting access to the secondary storage from an external node in the storage system shown in drawing 1.
- [Drawing 14] It is the block diagram showing the composition in the case of permitting access to the name server from an external node in the storage system shown in drawing 1.
- [Drawing 15] It is a composition **** block diagram in the case of permitting access via the redirector from an external node, and direct access in the storage system shown in drawing 1.
- [Drawing 16] It is a block diagram for explaining the conventional technique of realizing file sharing between two or more nodes (client) on a network.
- [Description of Notations]
- 1 Storage System
 - 2 Client
 - 3 External Network (Gigabit Ethernet)
 - 11 Redirector (Redirector, File Server Management Node)
 - 11a Gigabit Ethernet card

- 11b In FINI band card
- 11c Network processor
- 11d Memory (primary storage)
- 11e PCI (Peripheral Component Interconnect) bus
- 11f Cache memory (cache section)
- 12-1 - 12-n NFS (Network File System) server (file server)
- 12' Cache server
- 12a, 13a CPU (Central Processing Unit)
- 12b, 13b Memory (primary storage)
- 12c, 13c Interface card (IB-IF)
- 13 Name Server
- 14 High-speed (Interior) Network [in FINI Band (Infiniband)] Switch
- 15 Shared Memory (Shared Memory)
- 15-1 - 15-m Memory card (Shared Memory Card)
- 16 IB-FC Card
- 17 Secondary Storage
- 18 FC Switch
- 19 External Node
- 20 Network Disk Adapter
- 21 Header Unit
- 21a Physical-layer header (Phy Header)
- 21b IP header (Internet Protocol Header)
- 21c TCP header (Transmission Control Protocol Header)
- 21d NFS header
- 22 Real File Data Section
- 23 Header Offset Value (Boundary Information)
- 111 Request Analysis Section
- 111a Header offset value analysis section
- 111b Header offset pricing Kabe
- 112 Request Transfer Section
- 113 Transfer History Records Department
- 114 Load Monitoring Department
- 115 NFS Server Load Surveillance Demon (Daemon)
- 116 Response Section
- 121, 131 Taking over information Records Department
- 122 NIC (Network Interface Card) Driver (Network Driver)
- 123, 124 Buffer (mbuf)
- 125 File System Section
- 126 File System Buffer
- 132 NFS Server Surveillance Demon
- 133 Name Server Surveillance Demon
- 134 Malfunction Detection Section
- 135 Taking over Directions Section

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2002-163140

(P2002-163140A)

(43) 公開日 平成14年6月7日 (2002.6.7)

(51) Int.Cl.⁷

G 0 6 F 12/00

13/10

識別記号

5 4 5

3 4 0

F I

G 0 6 F 12/00

13/10

特記コード(参考)

5 4 5 B 5 B 0 1 4

3 4 0 A 5 B 0 8 2

審査請求 未請求 請求項の数 5 O L (全 16 頁)

(21) 出願番号

特願2000-359810(P2000-359810)

(22) 出願日

平成12年11月27日 (2000. 11. 27)

(71) 出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番
1号

(72) 発明者 大江 和一

神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

(72) 発明者 西川 克彦

神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

(74) 代理人 100092978

弁理士 真田 有

Fターム(参考) 5B014 EB04 FA05

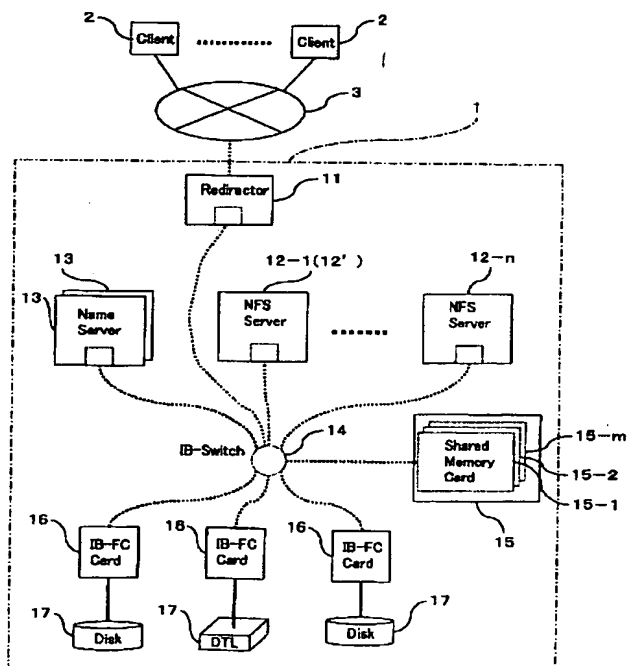
5B082 AA01 FA16 HA01 HA05 HA08

(54) 【発明の名称】 ストレージシステム

(57) 【要約】

【課題】 ネットワークの帯域拡大に対して低コストで十分に対応できるスケーラビリティをもったストレージシステムを提供することを目的とする。

【解決手段】 ファイルデータを記憶しうる記憶装置17と、ファイルデータに関するリクエストに応じたファイル処理を記憶装置17に対して行なう複数のファイルサーバ12-1~12-nと、外部ネットワーク3を介してクライアント2から受信されるファイルリクエストのファイルサーバ12-i (i=1~n) への転送処理と、そのファイルリクエストに対するクライアント2への応答処理とを管理するファイルサーバ管理ノード11と、記憶装置17、ファイルサーバ12-i 及びファイルサーバ管理ノード11を通信可能に相互接続する内部ネットワーク14とをそなえるように構成する。



【特許請求の範囲】

【請求項1】 ファイルデータを記憶しうる記憶装置と、
該ファイルデータに関するリクエストに応じたファイル処理を該記憶装置に対して行なう複数のファイルサーバと、
外部ネットワークを介してクライアントから受信されるリクエストの該ファイルサーバへの転送処理と、該リクエストに対する該クライアントへの応答処理とを一元管理するファイルサーバ管理ノードと、
該記憶装置、該ファイルサーバ及び該ファイルサーバ管理ノードを通信可能に相互接続する内部ネットワークとをそなえて構成されたことを特徴とする、ストレージシステム。

【請求項2】 該内部ネットワークに、該ファイルサーバが扱うファイルデータ名を一元管理するネームサーバが接続されていることを特徴とする、請求項1記載のストレージシステム。

【請求項3】 該内部ネットワークに、該ファイルサーバ管理ノード及び該ファイルサーバがアクセス可能な共有メモリが接続されていることを特徴とする、請求項1記載のストレージシステム。

【請求項4】 該内部ネットワークに、該ファイルサーバ管理ノード、該ファイルサーバ及び該ネームサーバがアクセス可能な共有メモリが接続されていることを特徴とする、請求項2記載のストレージシステム。

【請求項5】 該ファイルサーバ管理ノードが、該リクエストの内容を解析するリクエスト解析部と、該リクエスト解析部の解析結果に応じて該リクエストを特定のファイルサーバに転送するリクエスト転送部とをそなえていることを特徴とする、請求項1～4のいずれか1項に記載のストレージシステム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、ストレージシステムに関し、所望のネットワークに接続されることで複数のクライアントでファイルデータの共有を可能にするストレージシステムに関する。

【0002】

【従来の技術】 ネットワーク上での複数ノード（クライアント）間のファイルデータの共有（以下、単に、「ファイル共有」という）を実現する従来の手法としては、例えば図16に模式的に示すように、ネットワークファイルシステム（NFS：Network File System）を利用してLAN（Local Area Network）などの所望のネットワーク100上にファイルサーバ200を構築し、このファイルサーバ200にSCSI（Small Computer System Interface；一般に「スカジー」と呼ばれる）などのインタフェース300を介して二次記憶装置400を接続して、この二次記憶装置400において複数クライ

アント500間のファイル共有を実現する方法が良く知られている。

【0003】 しかしながら、この方法では、次のような課題がある。

①ファイルサーバを構築・維持（保守）するのに専門的なスキルが必要である。

②ファイルサーバの拡張（容量、アクセス性能）が容易でない。拡張できてもファイルサーバが複数に分かれたりしてしまうなどで維持コストが増大してしまう。

【0004】 ③故障時にそなえたシステム構築・維持（保守）に専門的なスキルが必要であり、また、そのための費用もかかる。

これらの課題を解決する方法として、近年、NAS（Network Attached Storage）が提案されている。このNASは、上記のファイルサーバ200及び二次記憶装置400から成る部分（図16中の破線枠参照）が予め1つのストレージシステムとして構築されたものに相当し、ネットワーク100に接続して簡単な設定を行なうだけで、ファイル共有が実現できるシステムで、システムの構築・維持（保守）に専門的なスキルは必要ない。

【0005】

【発明が解決しようとする課題】 しかしながら、このようなNASにおいても、現在急速に進んでいるLANの帯域拡大（現状で1Gbps、数年後には10Gbps程度）に低コストで十分に対応できるスケーラビリティが得られていないという課題は残っている。即ち、接続先のネットワークの帯域拡大に対応しようとすると、NASにおいても、単純に、内部のファイルサーバ及び二次記憶装置を増設することになり、この結果、ファイルサーバが複数に分かれてしまい、それぞれが管理する二次記憶装置も分割されてしまうことになる。

【0006】 つまり、上記のファイルサーバ200及び二次記憶装置400を並列（独立）して設けることになる。このため、結局、ファイルサーバの維持（保守）を個々に行なう必要があり、維持コストが増大してしまう。本発明は、このような課題に鑑み創案されたもので、ネットワークの帯域拡大に対して低コストで十分に対応できるスケーラビリティをもったストレージシステムを提供することを目的とする。

【0007】

【課題を解決するための手段】 上記の目的を達成するために、本発明のストレージシステム（請求項1）は、ファイルデータを記憶しうる記憶装置と、この記憶装置に対しリクエストに応じたファイル処理を行なう複数のファイルサーバと、外部ネットワークを介してクライアントから受信されるリクエストの上記ファイルサーバへの転送処理と、そのリクエストに対するクライアントへの応答処理とを一元管理するファイルサーバ管理ノードと、上記の記憶装置、ファイルサーバ及びファイルサーバ管理ノードを通信可能に相互接続する内部ネットワー

クとをそなえて構成されたことを特徴としている。

【0008】ここで、上記の内部ネットワークには、上記のファイルサーバが扱うファイルデータ名を一元管理するネームサーバが接続されていてもよい（請求項2）、上記のファイルサーバ管理ノード及びファイルサーバがアクセス可能な共有メモリが接続されていてもよい（請求項3）。なお、上記のネームサーバが存在する場合は、上記のファイルサーバ管理ノード及びファイルサーバに加えて、このネームサーバも上記の共有メモリにアクセスが可能である（請求項4）。

【0009】また、上記のファイルサーバ管理ノードは、例えば、上記のリクエストの内容を解析するリクエスト解析部と、このリクエスト解析部の解析結果に応じて該リクエストを特定のファイルサーバに転送するリクエスト転送部とをそなえて構成されるのが好ましい（請求項5）。

【0010】

【発明の実施の形態】以下、図面を参照して本発明の実施の形態を説明する。

（A）一実施形態の説明

図1は本発明の一実施形態としてのストレージシステムの構成（ストレージアーキテクチャ）を示すブロック図で、この図1に示すストレージシステム1（以下、単に「システム1」ともいう）は、外部ネットワーク（例えば、ギガビットイーサネット（登録商標））3に接続された複数のクライアント2間でファイル共有を実現するためのものであって、リダイレクタ（Redirector）11、複数のNFSサーバ（ファイルサーバ）12-1～12-n、ネームサーバ13、共有メモリ（Shared Memory）15、I B-F Cカード16、二次記憶装置17（ディスク装置やテープ装置（DTL）など）をそなえており、これらの各コンポーネント11、12-i、13、15、16、17が例えば4～10Gbps（ギガビット/秒）程度の高速（内部）ネットワーク（インフィニバンド（Infiniband））スイッチ14を介して相互に接続された構成になっている。

【0011】このようにシステム内部の各コンポーネント（リダイレクタ11、NFSサーバ12-i、ネームサーバ13、二次記憶装置17など）を内部ネットワーク14で接続する形態（クラスタリング）をとることで、必要に応じてNFSサーバ12-iや二次記憶装置17を内部ネットワーク14に接続すれば簡単に増設することができ、外部ネットワーク3の通信速度などに応じた拡張性（容量、アクセス性能など）が大幅に向上する。

【0012】ここで、上記のリダイレクタ（ファイルサーバ管理ノード）11は、外部ネットワーク3を介して任意のクライアント2から受信される各種リクエストメッセージ（以下、単に「リクエスト」という）のNFSサーバ12-i（ $i=1\sim n$ ）への転送処理と、そのリ

クエストに対するリクエスト元のクライアント2への応答処理とを一元（集中）管理するためのものである。つまり、本リダイレクタ11の存在により、上述のごとくNFSサーバ12-iや二次記憶装置17を増設したとしても、従来のようにそれらを個々に保守する必要は無いのである。

【0013】なお、上記の「リクエスト」は、二次記憶装置17に記憶されたファイルデータ（以下、単に「ファイル」ともいう）に関する要求を意味し、例えば、ファイルデータの実体に対するファイル操作（書き込み／更新／読み出しなど）要求（ファイルアクセスリクエスト）や、それ以外のファイル名参照などのメタ情報アクセスなどがある。また、各クライアント2からは、基本的に、このリダイレクタ11に付与されたIP（Internet Protocol）アドレスのみが参照できる。つまり、各クライアント2からは、本システム1が1つのファイルサーバとして見えるようになっている。

【0014】そして、上述したような機能を実現すべく、本リダイレクタ11には、例えば図2に示すように、外部ネットワーク3用のインタフェースを装備するギガビットイーサネットカード11a、システム内部（内部ネットワーク）用のインタフェースを装備するインフィニバンドカード11b、これらの各カード11a、11bをはじめとして自身（リダイレクタ11）の動作を集中制御するためのネットワークプロセッサ11c、このネットワークプロセッサ11cが動作する上で必要なソフトウェア（プログラム）や各種データを記憶するためのメモリ（主記憶部）11dなどが実装される。

【0015】なお、ネットワークプロセッサ11cは、これらのコンポーネント11a～11dとPCI（Peripheral Component Interconnect）バス11eを介して相互通信可能に接続されている。ここで、上記のネットワークプロセッサ11cは、内部ネットワーク14と外部ネットワーク3との間で送受されるリクエストやそのリクエストに対するリプライ（応答）メッセージなどの送受（プロトコル変換なども含む）や、クライアント2から受信されるリクエスト（プロトコル）の解析、その解析結果に基づいたアクセスファイル名の決定、受信リクエストの転送先NFSサーバ12-iの決定などを行なうことができるもので、本実施形態では、次のような制御も可能になっている。

【0016】即ち、クライアント2からのリクエストを解析し、各ファイルサーバ12-iに均一に（例えば、負荷の軽いNFSサーバ12-iに）リクエストが転送されるように制御したり、同一ファイルについてのリクエストは同じNFSサーバ12-iに割り当てて、NFSサーバ12-i間でファイルアクセス競合が起きないように制御したりすることができるようになっている。

【0017】このため、本ネットワークプロセッサ11

cには、その要部の機能に着目すると、次のような機能が実装されている。

(1) クライアント2からのリクエストの内容を解析するリクエスト解析部111としての機能

(2) このリクエスト解析部111の解析結果に応じて受信リクエストを特定のNFSサーバ12-iに転送するリクエスト転送部112としての機能

(3) リクエスト転送部112による過去のリクエストの転送履歴を(例えば、メモリ11dに)記録する転送履歴記録部113としての機能

(4) NFSサーバ負荷監視デーモン(daemon)115をバックグラウンドで実行することにより、各NFSサーバ12-iの負荷状態を定期的に監視する負荷監視部114としての機能

これにより、上記のリクエスト転送部112は、上記の転送履歴記録部113による転送履歴に基づいて同一ファイル名のファイルに対するリクエストを同一NFSサーバ12-iに転送したり、NFSサーバ負荷監視デーモン115(負荷監視部114)による負荷監視結果に基づいて負荷の低いNFSサーバ12-iに受信リクエストを転送したりすることが可能になる。

【0018】従って、処理速度が大幅に向上するとともに、各NFSサーバ12-iに均一にリクエストを転送できるので、一部のNFSサーバ12-iに負荷が集中して障害を引き起こしてしまうようなことを確実に防止することができ、信頼性が大幅に向上する。なお、本ネットワークプロセッサ11cは、例えば、メタ情報アクセスに対するリプライメッセージを内部のキャッシュメモリ11f(メモリ11dでもよい)にキャッシュしておき(図3参照)、クライアント2からメタ情報アクセスについてのリクエストを受けると、まず、キャッシュメモリ11f(あるいは、メモリ11d)をチェックして、ヒットすればリプライメッセージを作成してそのまま(NFSサーバ12-iやネームサーバ13に転送することなく)クライアント2側に戻す(図4参照)ようにすることもできる。

【0019】この仕組みは、メタ情報のみならずファイルデータについても適用できる。ただし、この場合、全てのファイルデータをキャッシュするようにすると必要なメモリ容量が増大するので、ネットワークプロセッサ11cでアクセス頻度の高いファイルデータを選び出し、そのファイルデータのみをキャッシュメモリ11f(あるいは、メモリ11d)にキャッシュするようにした方がよい。

【0020】つまり、上記のキャッシュメモリ11f(あるいは、メモリ11d)は、リクエスト頻度の高い特定のファイルデータについてのクライアント2へのリプライメッセージをキャッシュしておくキャッシュ部としての機能も果たし、この場合のネットワークプロセッサ11cは、新たなリクエストが同じファイルについて

のものであると、キャッシュメモリ11f(あるいは、メモリ11d)にキャッシュしておいたリプライメッセージをクライアント2へ返す応答部116(図2参照)としての機能も有していることになる。

【0021】このようにして、アクセス頻度の高い情報(メタ情報、ファイルデータ)に関しては、リダイレクタ11側でキャッシュして、NFSサーバ12-iにリクエストを転送することなくリダイレクタ11で応答を返してしまえば、アクセス頻度が高い情報に対するクライアント2への応答速度が大幅に向上し、本システム1としての処理速度及び処理能力が飛躍的に向上することになる。

【0022】次に、上記のNFSサーバ12-iは、それぞれ、リダイレクタ11から転送されてきたリクエストに応じたファイル処理(書き込み/更新/読み出しなど)を、内部ネットワーク(内部ネットワークスイッチ)14経由で二次記憶装置17にアクセスして実施したり、そのファイル処理結果をリクエスト元のクライアント2への応答として送付するためのリプライメッセージを生成してリダイレクタ11に送信したりすることができるものである。

【0023】なお、NFSサーバ12-iは、いずれも、ハードウェア的には、例えば図5に示すように、CPU(Central Processing Unit)12a、メモリ(主記憶部)12b及び内部ネットワーク14とのインタフェース(プロトコル変換など)を装備するインタフェースカード(IBM-IF)12cをそなえて構成され、メモリ12bに記憶されたNFSサーバソフトウェア(プログラム)をCPU12aが読み取って動作することにより、上述したNFSサーバ12-iとしての機能が実現されるようになっている。

【0024】さて、ここで、これらのNFSサーバ12-iが、それぞれ独自にファイル名を管理すると、同じファイルデータ実体であるにも関わらず異なるNFSサーバ12-iで管理ファイル名が異なったり、逆に、異なるファイルデータ実体であるにも関わらず異なるNFSサーバ12-iで同じ管理ファイル名になったりといった不都合が生じ、NFSサーバ12-i間でファイルアクセス競合が生じる可能性がある。

【0025】このような不都合を解決するのが上記のネームサーバ13である。つまり、このネームサーバ13において、NFSサーバ12-iからのメタ情報アクセスを一元管理することで、全てのNFSサーバ12-iにおける管理ファイル名空間を1つにして、NFSサーバ12-i間でのファイルアクセス競合を回避するのである。従って、このネームサーバ13をそなえることにより、本システム1によるファイル共有の信頼性が大幅に向上することとなる。

【0026】なお、図1中に示すように、本ネームサーバ13には、故障などの異常発生(ダウン)にそなえて

現用と予備用とが存在する。また、これらのネームサーバ13についても、ハードウェア的には、NFSサーバ12-iと同様の構成(図5参照)、即ち、CPU13a、メモリ(主記憶部)13b及び内部ネットワーク14とのインタフェースを装備するインタフェースカード13cが実装されており、この場合も、メモリ13bに記憶されたネームサーバソフトウェア(プログラム)をCPU13aが読み取って動作することにより、上述したネームサーバ13としての機能が実現されるようになっている。

【0027】次に、上記の共有メモリ15は、上記のリダイレクタ11、NFSサーバ12-i及びネームサーバ13がそれぞれ内部ネットワーク14を介してアクセス可能なメモリで、例えば、或るNFSサーバ12-iあるいは現用のネームサーバ13がダウンした場合(障害発生時)に、そのNFSサーバ12-iあるいは現用のネームサーバ13の処理を他のNFSサーバ12-k($k=1\sim n$ で $k\neq i$)あるいは予備用のネームサーバ13に引継ぐための情報(以下、引継ぎ情報)などがサーバ12-i、13別にメモリカード(Shared Memory Card)15-1~15-m(mは自然数)に保持(バックアップ)されるようになっている(図6及び図8参照)。

【0028】つまり、上記の各NFSサーバ12-i(CPU12a)もしくはネームサーバ13(CPU13a)は、異常発生時にそなえて他のNFSサーバ12-iもしくは予備用のネームサーバ13に処理を引継ぐのに必要な情報を引継ぎ情報として共有メモリ15に記録する引継ぎ情報記録部121(131)(図5参照)としての機能を有していることになる。

【0029】なお、NFSサーバ12-iのダウンは、例えば図6に模式的に示すように、現用のネームサーバ13(CPU13a)がNFSサーバ監視デーモン132をバックグラウンドで実行することで監視し、現用のネームサーバ13のダウンは、例えば図8に模式的に示すように、予備用のネームサーバ13(CPU13a)がネームサーバ監視デーモン133をバックグラウンドで実行することで監視する。

【0030】そして、図7に模式的に示すように、NFSサーバ12-iのダウンが検出された場合(ステップS1)は、現用のネームサーバ13(CPU13a)が、ダウンしたNFSサーバ12-i以外のNFSサーバ12-k(例えば、負荷の軽いNFSサーバ12-k)に対して、ダウンしたNFSサーバ12-iの処理を引継ぐよう指示するとともに、リダイレクタ11に対してNFSサーバ12-iのダウンを通知する(ステップS2)。

【0031】これにより、引継ぎ指示を受けたNFSサーバ12-k(CPU12a)は、共有メモリ15に内部ネットワーク14を介してアクセスして、そこにバック

アップされている引継ぎ情報を読み出してダウンしたNFSサーバ12-iの処理を引継ぐ(ステップS3)。一方、このとき、リダイレクタ11(ネットワークプロセッサ11c)は、ネームサーバ13から上記の通知を受けることにより、ダウンしたNFSサーバ12-iにはリクエスト転送部112によるリクエストの転送を行なわないようにする。

【0032】つまり、この場合のネームサーバ13(CPU13a)は、NFSサーバ12-iの異常発生を検出する異常検出部134(図6参照)と、この異常検出部134でNFSサーバ12-iの異常発生が検出されると、そのNFSサーバ12-i以外の他のNFSサーバ12-kに対して異常発生したNFSサーバ12-iの処理を共有メモリ15の引継ぎ情報に基づいて引継ぐよう引継ぎ指示を与える引継ぎ指示部135(図6参照)としての機能を有していることになる。

【0033】このように、本実施形態では、たとえ、NFSサーバ12-iやネームサーバ13がダウンしたとしても、他のNFSサーバ12-kや予備用のネームサーバ13が処理を引継ぐことができるので、ストレージシステム1としては正常なファイル処理を継続することができ、耐障害性が大幅に向上する。なお、本例は、NFSサーバ12-iやネームサーバ13の冗長化についての説明であるが、勿論、リダイレクタ11を同様にして冗長化することも可能である。また、現用のネームサーバ13ダウン時の引継ぎについては、場合によってはNFSサーバ12-iのいずれかに引継ぐようにしてもよい。

【0034】次に、上述したリダイレクタ11からNFSサーバ12-iへのリクエスト転送時のリダイレクタ11及びNFSサーバ12-iでの具体的な処理について説明する。リダイレクタ11は、例えば、クライアント2から或るファイルデータを書き込むためのファイルアクセスリクエストを受信すると、そのファイルアクセスリクエストをリクエスト解析部111にて解析する。

【0035】なお、上記の「ファイルアクセスリクエスト」は、例えば図12に示すように、物理レイヤヘッダ(Phy Header)21aやIPヘッダ(Internet Protocol Header)21b、TCPヘッダ(Transmission Control Protocol Header)21c、NFSヘッダ21dなどから成るヘッダ部21と、実際に二次記憶装置17に書き込むべきファイルデータの实体(実ファイルデータ)が格納された実ファイルデータ部22とを有して成る。

【0036】そして、リクエスト解析部111は、図9に模式的に示すように、上記のファイルアクセスリクエスト中の実ファイルデータが始まる位置、即ち、ヘッダ部21と実ファイルデータ部22との境界をヘッダオフセット値[境界情報;例えば先頭からのビット数("a")など]23として求める。求められた境界情報23は、リクエスト転送部112に通知され、リクエ

スト転送部112は、上記ファイルアクセスリクエストに通知された境界情報23を添付（付加）して転送先のNFSサーバ12-iに送付する。

【0037】つまり、上記のリクエスト解析部111は、図2中に示すように、受信したファイルアクセスリクエストを解析してそのファイルアクセスリクエストのヘッダ部21と実ファイルデータ部22との境界位置を表わすヘッダオフセット値23を求めるヘッダオフセット値解析部111aとしての機能と、このヘッダオフセット値解析部111aで得られたヘッダオフセット値23をNFSサーバ12-iへ転送されるファイルアクセスリクエストに付加するヘッダオフセット値付加部111bとしての機能とを有していることになる。

【0038】その後、NFSサーバ12-i側では、上述のごとくリダイレクタ11側で付加されたヘッダオフセット値23に基づいて、NIC（Network Interface Card）ドライバ（ネットワークドライバ）122が、実ファイルデータ部22とそれ以外の領域（ヘッダ部21）の先頭アドレスをそれぞれ上位層（NFS処理層）のカーネル内（カーネル上位層）で扱われるメッセージのページ境界（ページ境界（別領域）；バッファ（mbuf）123、124）に割り当てることができる（図10参照）。

【0039】このようにすると、例えば図11に模式的に示すように、ファイルアクセスリクエストがカーネル上位層のファイルシステム部125に到達したときに、実ファイルデータ部22の先頭アドレス（ポインタ）をファイルシステムバッファ126へのポインタに付け替えるだけで、データのコピーを発生させずに（マップ切り替え）、ファイルシステムバッファ126にデータを移すことが可能になる（カーネル内ゼロコピーが実現される）。従って、DMA（Direct Memory Access）も高速に実行でき、NFSサーバ12-iの処理速度及び処理能力を大幅に向上することができる。

【0040】なお、ヘッダ部21と実ファイルデータ部22との境界は、NICドライバ122側で求めるようにすることも考えられるが、この場合、NICドライバ122の処理（ヘッダ解析）量が増大する（通常、NICドライバ122は物理レイヤヘッダ21aの解析のみを行なう）ため、上記のように元々ヘッダ部21の解析機能（リクエスト解析部111）を有するリダイレクタ11側で境界を求めるようにした方が、NICドライバ層での処理量を増やさずに（処理能力低下を招かず）上位層（NFS処理層）でのカーネルゼロコピーを実現できる。

【0041】以上のように、本実施形態のストレージシステム1によれば、システム1内部に、リダイレクタ11、複数のNFSサーバ12-i、ネームサーバ13、共有メモリ15、二次記憶装置17を設け、これらを高速な内部ネットワーク14で接続する構成にしたことに

より、必要に応じてNFSサーバ12-iや二次記憶装置17を簡単に増設することができ、しかも、NFSサーバ12-iの維持（保守）を個々に行なう必要もないので、外部ネットワーク3の帯域拡大に対して低コストで十分に対応できる（例えば、10Gbps LANまで対応可能な）性能・容量スケーラビリティを確保することができる。

【0042】特に、上述した実施形態では、リダイレクタ11が、各NFSサーバ12-iの負荷に応じて各NFSサーバ12-iに均一に処理が割り当てられるように制御したり、同じファイルについてのリクエストに対する処理は同じNFSサーバ12-iに割り当てたり、アクセス頻度の高いファイルについてのリクエストに対してはNFSサーバ12-i側ではなくリダイレクタ11側でキャッシュしておいたりプライメッセージにより応答したりするので、その処理速度及び性能が飛躍的に向上しており、10Gbps LANまで確実に対応可能な性能・容量スケーラビリティが実現されている。

【0043】（B）第1変形例の説明

上述したシステム1内には、メモリ12bの容量が他のNFSサーバ12-iよりも大きいNFSサーバ12-iをキャッシュサーバ12'（図1参照）として配置して、このキャッシュサーバ12'でのファイルアクセスは基本的にメモリ12bに対する読み書きのみでクライアント2側に応答を返すようにしてもよい。

【0044】そして、一定期間内のリクエスト頻度が所定回数（閾値）以上であるアクセス頻度の高いファイルについてはキャッシュサーバ12'のメモリ12bにキャッシュしておき、キャッシュサーバ12'が応答するようにする。具体的には、まず、リダイレクタ11側で各ファイルのアクセス頻度を監視しておき、アクセス頻度が或る閾値よりも高いファイルへのアクセスに関しては、キャッシュサーバ12-iで処理を行なうように、リダイレクタ12-iからネームサーバ13、NFSサーバ12-i、キャッシュサーバ12'に指示を与える。

【0045】つまり、このとき、リダイレクタ11（リクエスト転送部112）は、アクセス頻度の高いファイルについてのリクエストをキャッシュサーバ12'に転送することになる。これにより、アクセス頻度の高いファイルに対するアクセスは、二次記憶装置17にアクセスすることなくキャッシュサーバ12'内で処理されるので、ストレージシステム1としての処理速度及び処理能力の大幅な向上に大きく寄与する。

【0046】一方、上記のアクセス頻度の高いファイルに対するアクセス頻度が落ちてくると（キャッシュサーバ12'のメモリ12bにキャッシュされているファイルに対するアクセス頻度が所定回数以下になると）、リダイレクタ11が適当な（例えば、負荷の軽い）NFSサーバ12-iを割り当てて処理を移行するようにネー

ムサーバ13, NFSサーバ12-i, キャッシュサーバ12'に指示する。

【0047】つまり、この場合、リダイレクタ11(リクエスト転送部112)は、リクエストの転送先をキャッシュサーバ12'以外のNFSサーバ12-iに変更するのである。これにより、アクセス頻度が落ちてきたファイルがいつまでもキャッシュサーバ12'にキャッシュされ続けることが回避され、その結果、キャッシュサーバ12'に必要なメモリ容量を削減することができるとともに、キャッシュサーバ12'での処理に余裕をもたせてその処理能力を向上することができる。

【0048】(C)第2変形例の説明

なお、例えば図13に示すように、上記の二次記憶装置17をFCスイッチ18経由でネームサーバ13及びNFSサーバ12-iと接続(二次記憶ネットワークを構築)し、FCスイッチ18と外部ノード19とを接続することにより、二次記憶装置17に外部ノード19からもアクセスできるようにすることも可能である。ただし、この場合、外部ノード19で動作させるファイルシステムは、ストレージシステム1内のファイルシステムと同じもの(上述した例では、NFS;図13中の網かけ部がこれを意味する)である必要がある。

【0049】このようにすることで、システム1内部のNFSサーバ12-iとのアクセス競合を避けるために外部ノード19からのアクセスについては何らかの調停制御が必要になるが、外部ノード19から本ストレージシステム1内のファイルへのアクセスが可能になる。ただし、例えば図14に示すように、外部ノード19からネームサーバ13へのアクセスを許容するようにすれば、外部ノード19からのアクセスについてもシステム1内の管理ファイル名に従うことになるので、上記の調停制御を必要とせず外部ノード19からのファイルアクセスが行なえるようになる。なお、この図14では、内部ネットワーク14経由でのネームサーバ13へのアクセスを許容する場合であるが、勿論、図13において二次記憶ネットワーク(FCスイッチ18)経由でのアクセスを許容するようにしてもよい。

【0050】また、例えば図15に示すように、NFSサーバ12-iと外部ノード19とを共通化してしまってもよい。つまり、NFSサーバ12-iを、外部ネットワーク3から直接受信されるリクエストに応じたファイル処理を二次記憶装置17に対して行なうように構成するのである。これにより、或るクライアント2がリダイレクタ11経由でアクセスしてきた場合は、NFSサーバ12-iは上述したストレージシステム1のファイルサーバとして機能し、直接、NFSサーバ12-iにアクセスしてきた場合は、リダイレクタ11を経由せずに応答する通常のファイルサーバとして機能することになる。つまり、クライアント2からのNFSサーバ12-i経由でのアクセスとNFSサーバ12-iを経由し

ない直接アクセスとの双方を許容するのである。

【0051】いずれの場合も、外部からの直接アクセスが可能となり、他のストレージアーキテクチャ(例えば、SAN(Storage Area Network)など)との融合を実現することができる。なお、図14及び図15において、符号20は内部ネットワーク14と二次記憶装置17とのインタフェースを装備するネットワークディスクアダプタを表す。

【0052】また、図13~図15に示す構成では、前述した共有メモリ15が省略されているが、勿論、装備されていてもよい。このようにすれば、図13~図15に示す構成においても、前記と同様のバックアップ処理が可能になる。

(D)その他

なお、上述した実施形態では、内部ネットワーク14としてInfiniband、外部ネットワーク3としてギガビットイーサネットを適用した場合について説明したが、勿論、これら以外の他の高速ネットワークを用いてシステム構築することも可能である。

【0053】また、上記のネームサーバ13や共有メモリ15は必ずしもそなえる必要はなく、これらのいずれか、あるいは、双方を省略しても本発明の目的は十分達成される。さらに、上述した実施形態では、ファイルサーバにNFSを適用しているが、本発明はこれに限定されず、勿論、他のファイルシステムを適用することも可能である。

【0054】また、上述した実施形態では、内部ネットワーク14の容量(通信速度)が4~10Gbps程度であることを前提としたが、この速度は外部ネットワーク3の帯域拡大に応じて適宜変更すれば対応できる。そして、本発明は、上述した実施形態に限定されず、上記以外にも本発明の趣旨を逸脱しない範囲で種々変形して実施することができる。

【0055】(E)付記

〔付記1〕 ファイルデータを記憶しうる記憶装置と、リクエストに応じたファイル処理を該記憶装置に対して行なう複数のファイルサーバと、外部ネットワークを介してクライアントから受信されるリクエストの該ファイルサーバへの転送処理と、該リクエストに対する該クライアントへの応答処理とを一元管理するファイルサーバ管理ノードと、該記憶装置、該ファイルサーバ及び該ファイルサーバ管理ノードを通信可能に相互接続する内部ネットワークとをそなえて構成されたことを特徴とする、ストレージシステム。

【0056】〔付記2〕 該内部ネットワークに、該ファイルサーバが扱うファイルデータ名を一元管理するネームサーバが接続されていることを特徴とする、付記1記載のストレージシステム。

〔付記3〕 該内部ネットワークに、該ファイルサーバ管理ノード及び該ファイルサーバがアクセス可能な共有

メモリが接続されていることを特徴とする、付記1記載のストレージシステム。

【0057】〔付記4〕 該内部ネットワークに、該ファイルサーバ管理ノード、該ファイルサーバ及び該ネームサーバがアクセス可能な共有メモリが接続されていることを特徴とする、付記2記載のストレージシステム。

〔付記5〕 該ファイルサーバ管理ノードが、該リクエストの内容を解析するリクエスト解析部と、該リクエスト解析部の解析結果に応じて該リクエストを特定のファイルサーバに転送するリクエスト転送部とをそなえていることを特徴とする、付記1～4のいずれか1項に記載のストレージシステム。

【0058】〔付記6〕 該ファイルサーバ管理ノードが、該リクエスト転送部による過去のリクエストの転送履歴を記録する転送履歴記録部をそなえ、該リクエスト転送部が、該転送履歴記録部の該転送履歴に基づいて同一ファイルデータ名のファイルデータに対するリクエストを同一ファイルサーバに転送するように構成されたことを特徴とする、付記5記載のストレージシステム。

【0059】〔付記7〕 該ファイルサーバ管理ノードが、該ファイルサーバの負荷を監視する負荷監視部をそなえ、該リクエスト転送部が、該負荷監視部での監視結果に基づいて負荷の低いファイルサーバに該リクエストを転送するように構成されたことを特徴とする、付記5記載のストレージシステム。

【0060】〔付記8〕 該ファイルサーバのうち少なくとも1台が、該記憶装置におけるファイルデータをキャッシュする主記憶部をそなえ、該主記憶部において該リクエストに応じたファイル処理を実行するキャッシュサーバとして構成されていることを特徴とする、付記5記載のストレージシステム。

〔付記9〕 該キャッシュサーバの該主記憶部が、一定期間内のリクエスト頻度が所定回数以上のファイルデータをキャッシュしておくように構成されるとともに、該リクエスト転送部が、上記のリクエスト頻度の高いファイルデータについてのリクエストを該キャッシュサーバに転送するように構成されたことを特徴とする、付記8記載のストレージシステム。

【0061】〔付記10〕 該リクエスト転送部が、該キャッシュサーバの該主記憶部にキャッシュされている該ファイルデータに対するリクエスト頻度が所定回数以下になると、該リクエストの転送先を該キャッシュサーバ以外のファイルサーバに変更するように構成されたことを特徴とする、付記9記載のストレージシステム。

【0062】〔付記11〕 該リクエスト解析部が、該リクエストを解析して当該リクエストのヘッダ部と実ファイルデータ部との境界位置を表わすヘッダオフセット値を求めるヘッダオフセット値解析部と、該ヘッダオフセット値解析部で得られた該ヘッダオフセット値を該ファイルサーバへ転送される該リクエストに付加するヘッ

ダオフセット値付加部とをそなえていることを特徴とする、付記5記載のストレージシステム。

【0063】〔付記12〕 該ファイルサーバが、該リクエストに付加された該ヘッダオフセット値に基づいて該リクエストの該ヘッダ部と該データ部とをそれぞれカーネル上位層で扱われるメッセージの異なる領域にコピーするネットワークドライバをそなえていることを特徴とする、付記11記載のストレージシステム。

〔付記13〕 該ファイルサーバ管理ノードが、リクエスト頻度の高い特定のファイルデータについての該クライアントへの応答メッセージをキャッシュしておくキャッシュ部と、該リクエストが該特定のファイルデータについてのものであると、該キャッシュ部の該当応答メッセージを該クライアントへ返す応答部とをそなえていることを特徴とする、付記1～12のいずれか1項に記載のストレージシステム。

【0064】〔付記14〕 該ファイルサーバが、異常発生時にそなえて他のファイルサーバに処理を引継ぐのに必要な情報を引継ぎ情報として該共有メモリに記録する引継ぎ情報記録部をそなえていることを特徴とする、付記3又は付記4に記載のストレージシステム。

〔付記15〕 該ファイルサーバの異常発生を検出する異常検出部と、該異常検出部で該ファイルサーバの異常発生が検出されると、当該ファイルサーバ（以下、異常ファイルサーバという）以外の他のファイルサーバに対して該異常ファイルサーバの処理を該共有メモリの該引継ぎ情報に基づいて引継ぐよう引継ぎ指示を与える引継ぎ指示部とが設けられたことを特徴とする、付記14記載のストレージシステム。

【0065】〔付記16〕 該記憶装置が、外部ノードからのアクセスを許容するように構成されたことを特徴とする、付記1～15のいずれか1項に記載のストレージシステム。

〔付記17〕 該ネームサーバが、外部ノードからのアクセスを許容するように構成されたことを特徴とする、付記2～15のいずれか1項に記載のストレージシステム。

【0066】〔付記18〕 該ファイルサーバが、該外部ネットワークから直接受信されるリクエストに応じたファイル処理を該記憶装置に対して行なうように構成されたことを特徴とする、付記1～17のいずれか1項に記載のストレージシステム。

【0067】

【発明の効果】以上詳述したように、本発明のストレージシステムによれば、リクエストに応じたファイル処理を記憶装置に対して行なう複数のファイルサーバと、各ファイルサーバの処理を一元管理するファイルサーバ管理ノードと、記憶装置、ファイルサーバ及びファイルサーバ管理ノードを通信可能に相互接続する内部ネットワークとをそなえているので、必要に応じてファイルサー

バや記憶装置を簡単に増設することができ、しかも、各ファイルサーバの維持（保守）を個々に行なう必要もないので、外部ネットワークの帯域拡大に対して低コストで十分に対応できる性能・容量スケーラビリティを確保することができる。

【0068】そして、上記の内部ネットワークに、上記の各ファイルサーバが扱うファイルデータ名を一元管理するネームサーバを接続すれば、ファイルサーバ間でファイルアクセス競合が生じることを防止することができるので、ファイル共有の信頼性向上に大きく寄与する。また、上記の内部ネットワークに、共有メモリを接続して、この共有メモリに障害発生時にそなえたファイルサーバの引継ぎ情報を随時記憶するようにしておくことで、一部のファイルサーバに障害が発生したとしても、ストレージシステムとしては正常な処理を継続することができるので、耐障害性が大幅に向上する。

【0069】さらに、ファイルサーバ管理ノードは、過去のリクエストの転送履歴に基づいて同一ファイルデータ名のファイルデータに対するリクエストを同一ファイルサーバに転送するように構成してもよく、このようにすれば、処理速度が大幅に向上する。また、ファイルサーバの負荷を監視して負荷の低いファイルサーバにリクエストを転送するようにしてもよく、このようにすれば、各ファイルサーバに均一にリクエストを転送できるので、一部のファイルサーバに負荷が集中して障害を引き起こしてしまうようなことを確実に防止することができる。信頼性が大幅に向上する。

【0070】さらに、リクエスト頻度が高いファイルデータについては、キャッシュサーバの主記憶部にキャッシュしておき、キャッシュサーバで処理するようにしておけば、記憶装置へのアクセス頻度を大幅に削減することができるので、さらに処理速度及び処理性能が向上する。そして、この場合、キャッシュサーバの主記憶部にキャッシュされているファイルデータに対するリクエスト頻度が所定回数以下になると、キャッシュサーバ以外のファイルサーバで処理を行なうようにすれば、リクエスト頻度の低いファイルデータがいつまでもキャッシュサーバに保持され続けることが回避されるので、キャッシュサーバの主記憶部に必要なメモリ容量を削減することができる。とともに、キャッシュサーバでの処理に余裕をもたせてその処理能力を向上することができる。

【0071】また、ファイルサーバ管理ノードにおいて、リクエストのヘッダ部と実ファイルデータ部との境界位置を表わすヘッダオフセット値を求めて、そのヘッダオフセット値をリクエストに付加してファイルサーバに転送するようにすれば、ファイルサーバのネットワークドライバにおいて、そのヘッダオフセット値に基づいてリクエストのヘッダ部とデータ部とをそれぞれカーネル上位層で扱われるメッセージの異なる領域にコピーすることができる。従って、カーネル内ゼロコピーを実現

することができ、ファイルサーバの処理速度及び処理能力を大幅に向上することができる。

【0072】さらに、上記のファイルサーバ管理ノードにおいて、リクエスト頻度の高い特定のファイルデータについてのクライアントへの応答メッセージをキャッシュしておき、受信したリクエストがそのファイルデータについてのものであると、キャッシュしておいた応答メッセージをクライアントへ返すようにすれば、ファイルサーバへリクエストを転送する必要が無いので、クライアントに対する応答速度が大幅に向上し、本システムとしての処理速度及び処理能力が飛躍的に向上することになる。

【0073】また、上記の記憶装置は、外部ノードからのアクセスを許容するように構成してもよく、このようにすれば、他のストレージアーキテクチャとの融合が可能になる。ここで、外部ノードから上記のネームサーバへのアクセスを許容するようにすれば、上記のファイルサーバと外部ノードとのファイルアクセス調停制御を必要とせずに、外部ノードからのファイルアクセスが可能になる。

【0074】さらに、上記のファイルサーバは、上記の外部ネットワークから直接受信されるリクエストに応じたファイル処理を記憶装置に対して行なうように構成してもよく、このようにすれば、クライアントからのファイルサーバ経由でのアクセスとファイルサーバを経由しない直接アクセスとの双方を許容することができるので、この場合も、他のストレージアーキテクチャとの融合が可能になる。

【図面の簡単な説明】

【図1】本発明の一実施形態としてのストレージシステムの構成（ストレージアーキテクチャ）を示すブロック図である。

【図2】図1に示すリダイレクタの構成を示すブロック図である。

【図3】図1に示すリダイレクタでメタ情報をキャッシュする場合を説明するためのブロック図である。

【図4】図1に示すリダイレクタでリプライメッセージを返す場合を説明するためのブロック図である。

【図5】図1に示すNFSサーバ（ネームサーバ）の構成を示すブロック図である。

【図6】図1に示す共有メモリにNFSサーバの引継ぎ情報をバックアップする場合を説明するためのブロック図である。

【図7】図6に示す共有メモリにバックアップされた引継ぎ情報に基づいてダウンしたNFSサーバの処理を引継ぐ場合を説明するためのブロック図である。

【図8】図1に示す共有メモリにネームサーバの引継ぎ情報をバックアップする場合を説明するためのブロック図である。

【図9】図1に示すリダイレクタにおいてファイルアク

セスリクエストの境界情報を求める場合を説明するためのブロック図である。

【図10】図1に示すNFSサーバにおいて図9に示すファイルアクセスリクエストの境界情報に基づいてカーネル内ゼロコピーを実現する場合を説明するためのブロック図である。

【図11】図1に示すNFSサーバにおいて図9に示すファイルアクセスリクエストの境界情報に基づいてカーネル内ゼロコピーを実現する場合を説明するためのブロック図である。

【図12】図9～図11に示すファイルアクセスリクエストのフォーマット例を示す図である。

【図13】図1に示すストレージシステムにおいて外部ノードからの二次記憶装置へのアクセスを許容する場合の構成を示すブロック図である。

【図14】図1に示すストレージシステムにおいて外部ノードからのネームサーバへのアクセスを許容する場合の構成を示すブロック図である。

【図15】図1に示すストレージシステムにおいて外部ノードからのリダイレクト経由のアクセスと直接アクセスとを許容する場合の構成示すブロック図である。

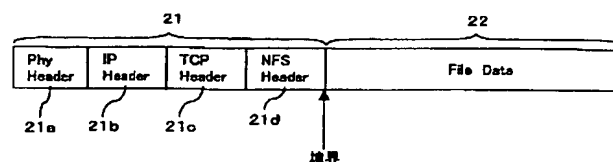
【図16】ネットワーク上での複数ノード（クライアント）間のファイル共有を実現する従来の手法を説明するためのブロック図である。

【符号の説明】

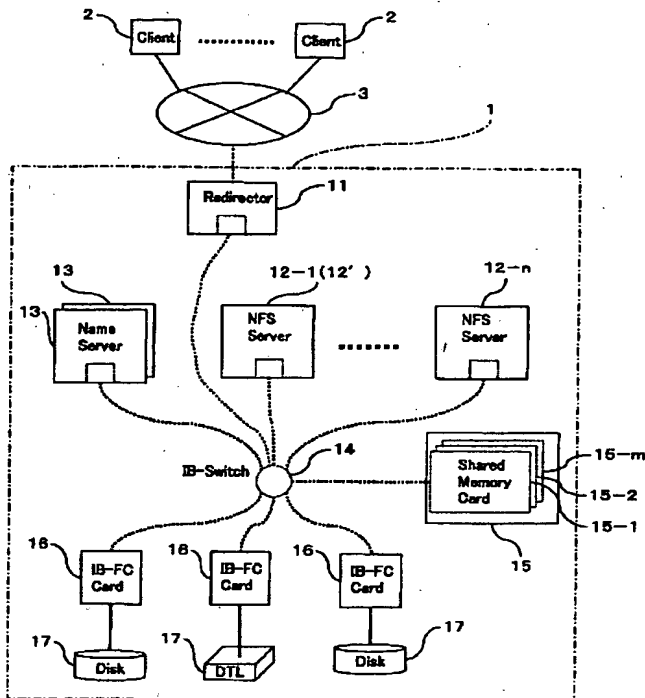
- 1 ストレージシステム
- 2 クライアント
- 3 外部ネットワーク（ギガビットイーサネット）
- 11 リダイレクタ（Redirector；ファイルサーバ管理ノード）
- 11a ギガビットイーサネットカード
- 11b インフィニバンドカード
- 11c ネットワークプロセッサ
- 11d メモリ（主記憶部）
- 11e PCI（Peripheral Component Interconnect）バス
- 11f キャッシュメモリ（キャッシュ部）
- 12-1～12-n NFS（Network File System）サーバ（ファイルサーバ）
- 12' キャッシュサーバ

- 12a, 13a CPU（Central Processing Unit）
- 12b, 13b メモリ（主記憶部）
- 12c, 13c インタフェースカード（IB-IF）
- 13 ネームサーバ
- 14 高速（内部）ネットワーク〔インフィニバンド（Infiniband）〕スイッチ
- 15 共有メモリ（Shared Memory）
- 15-1～15-m メモリカード（Shared Memory Card）
- 16 IB-FCカード
- 17 二次記憶装置
- 18 FCスイッチ
- 19 外部ノード
- 20 ネットワークディスクアダプタ
- 21 ヘッダ部
- 21a 物理レイヤヘッダ（Phy Header）
- 21b IPヘッダ（Internet Protocol Header）
- 21c TCPヘッダ（Transmission Control Protocol Header）
- 21d NFSヘッダ
- 22 実ファイルデータ部
- 23 ヘッダオフセット値（境界情報）
- 111 リクエスト解析部
- 111a ヘッダオフセット値解析部
- 111b ヘッダオフセット値付加部
- 112 リクエスト転送部
- 113 転送履歴記録部
- 114 負荷監視部
- 115 NFSサーバ負荷監視デーモン（daemon）
- 116 応答部
- 121, 131 引継ぎ情報記録部
- 122 NIC（Network Interface Card）ドライバ（ネットワークドライバ）
- 123, 124 バッファ（mbuf）
- 125 ファイルシステム部
- 126 ファイルシステムバッファ
- 132 NFSサーバ監視デーモン
- 133 ネームサーバ監視デーモン
- 134 異常検出部
- 135 引継ぎ指示部

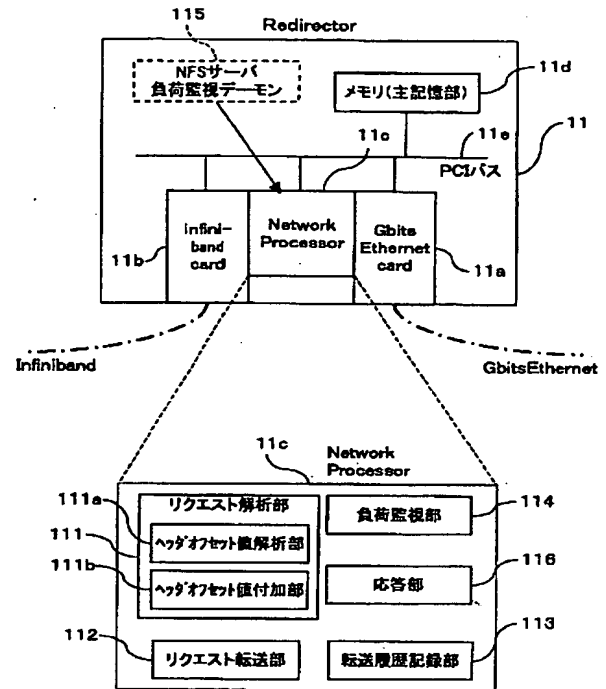
【図12】



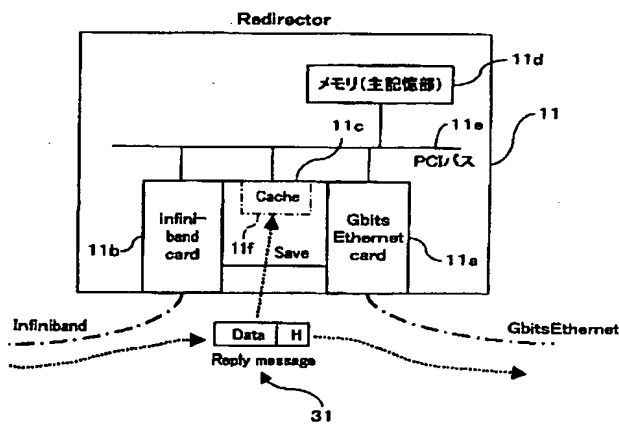
【図1】



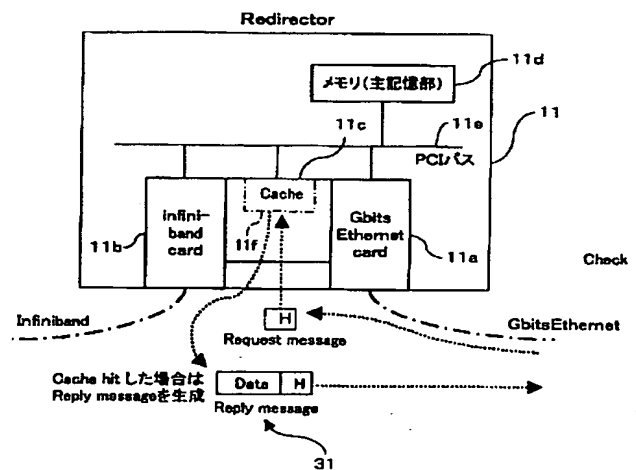
【図2】



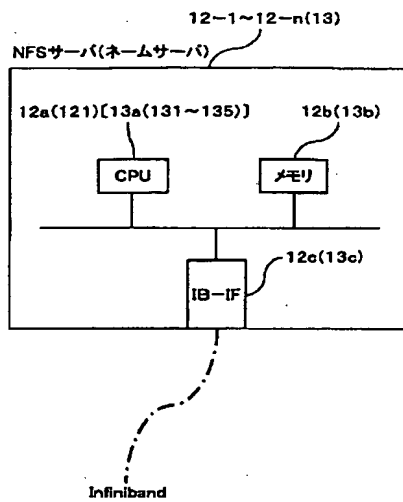
【図3】



【図4】

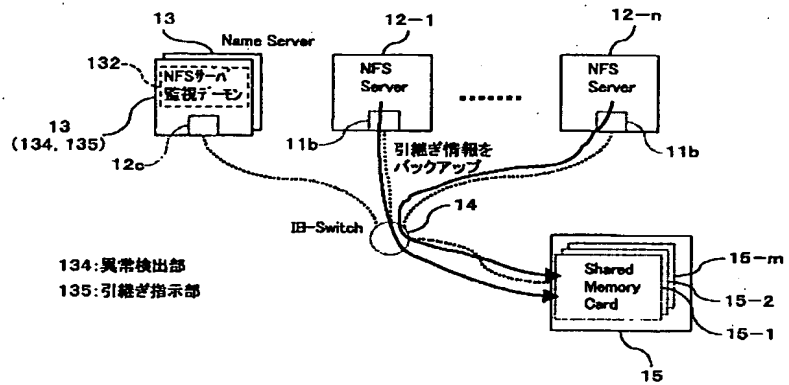


【図5】

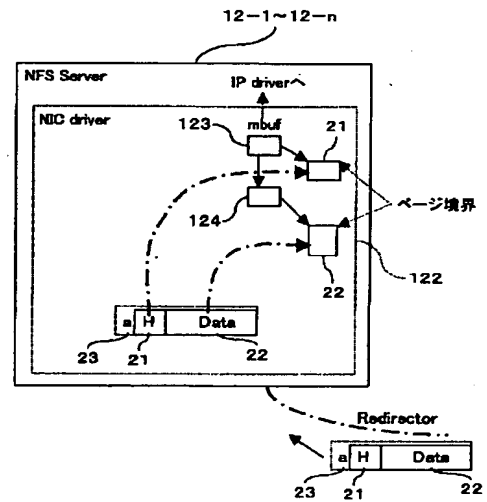


121, 131: 引継ぎ情報記録部
 132: NFSサーバ監視デモン
 133: ネームサーバ監視デモン
 134: 異常検出部
 135: 引継ぎ指示部

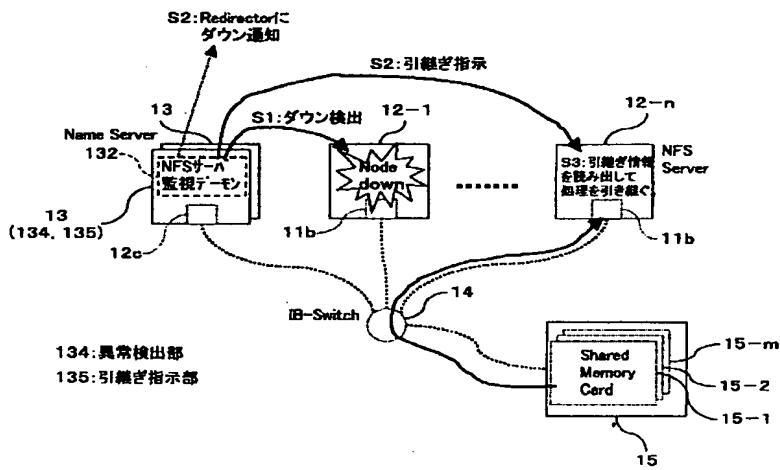
【図6】



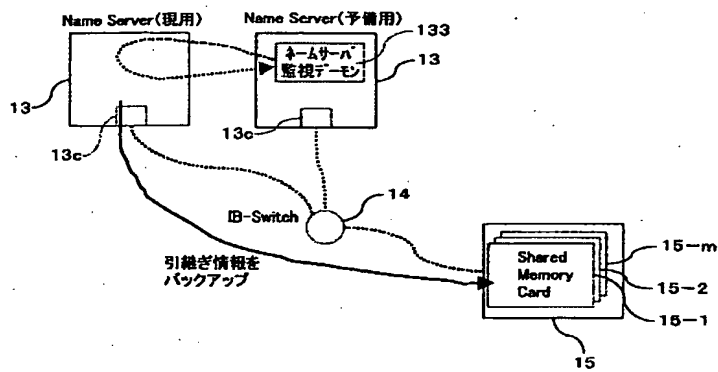
【図10】



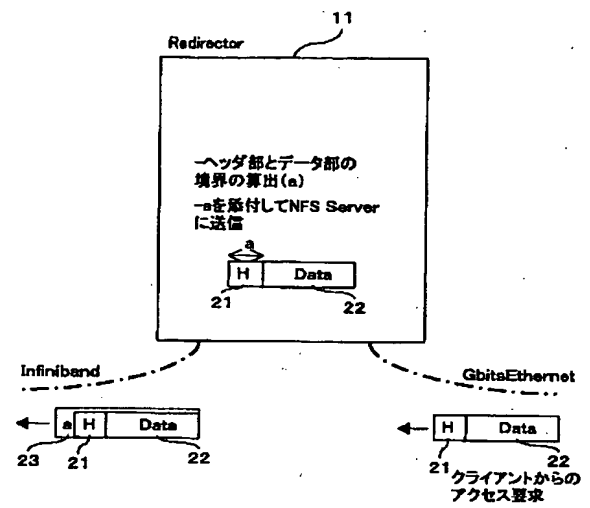
【図7】



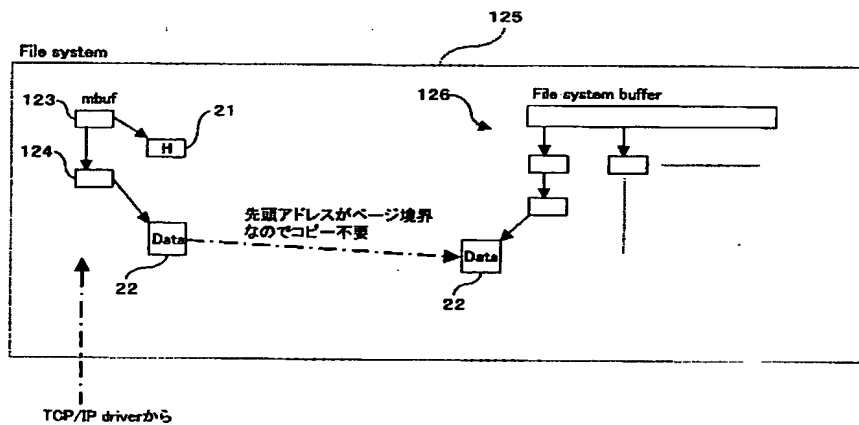
【図8】



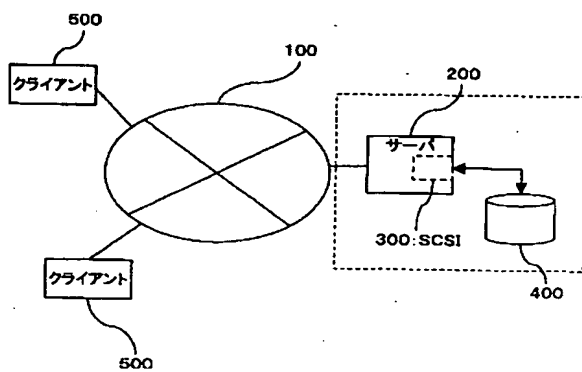
【図9】



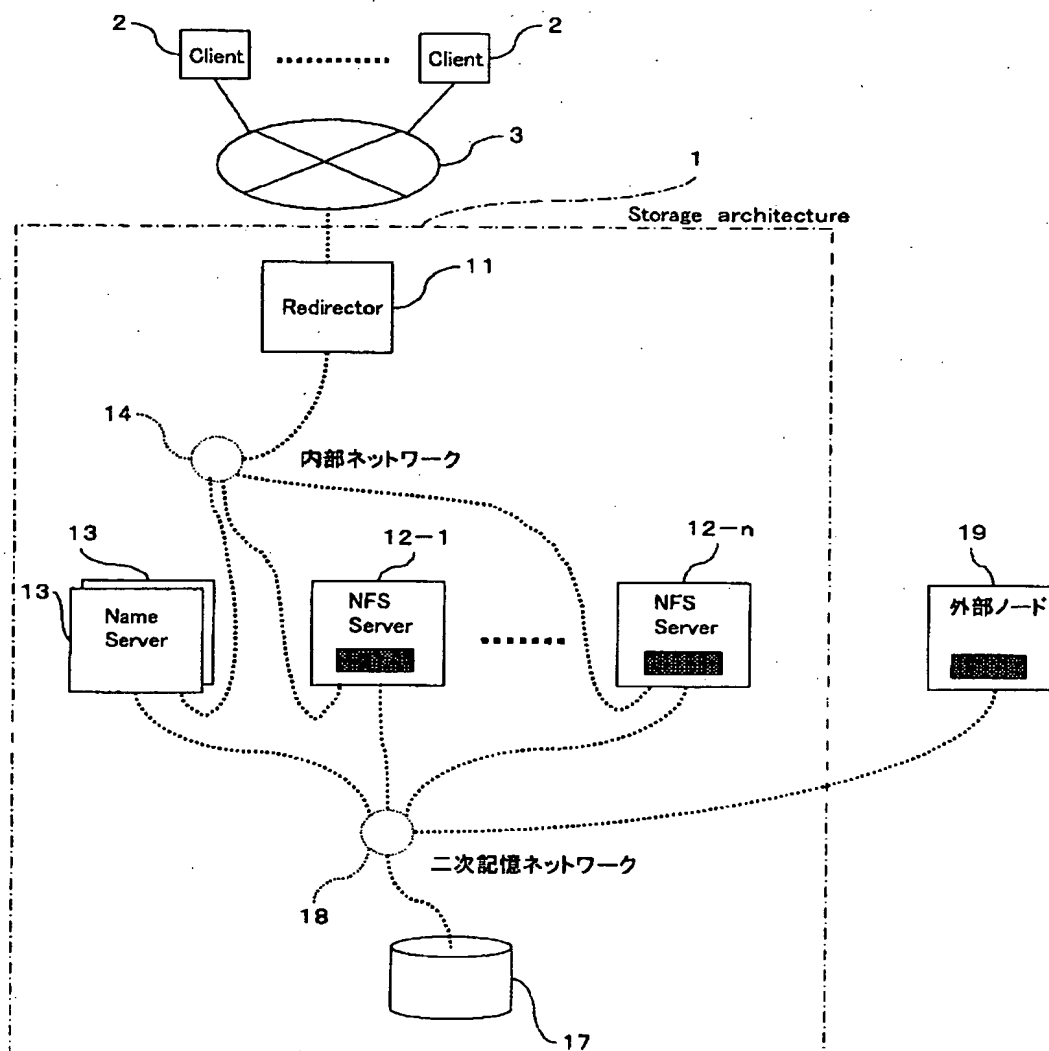
【図11】



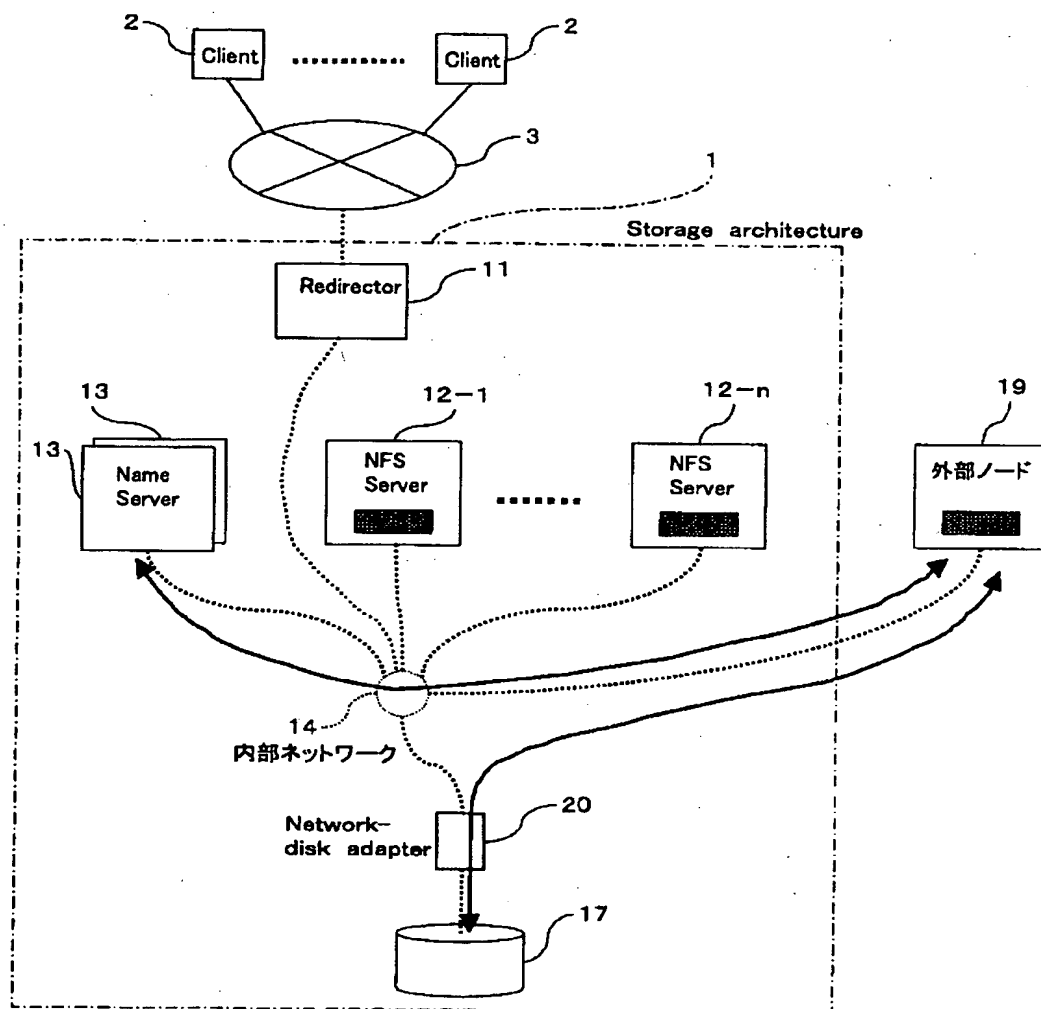
【図16】



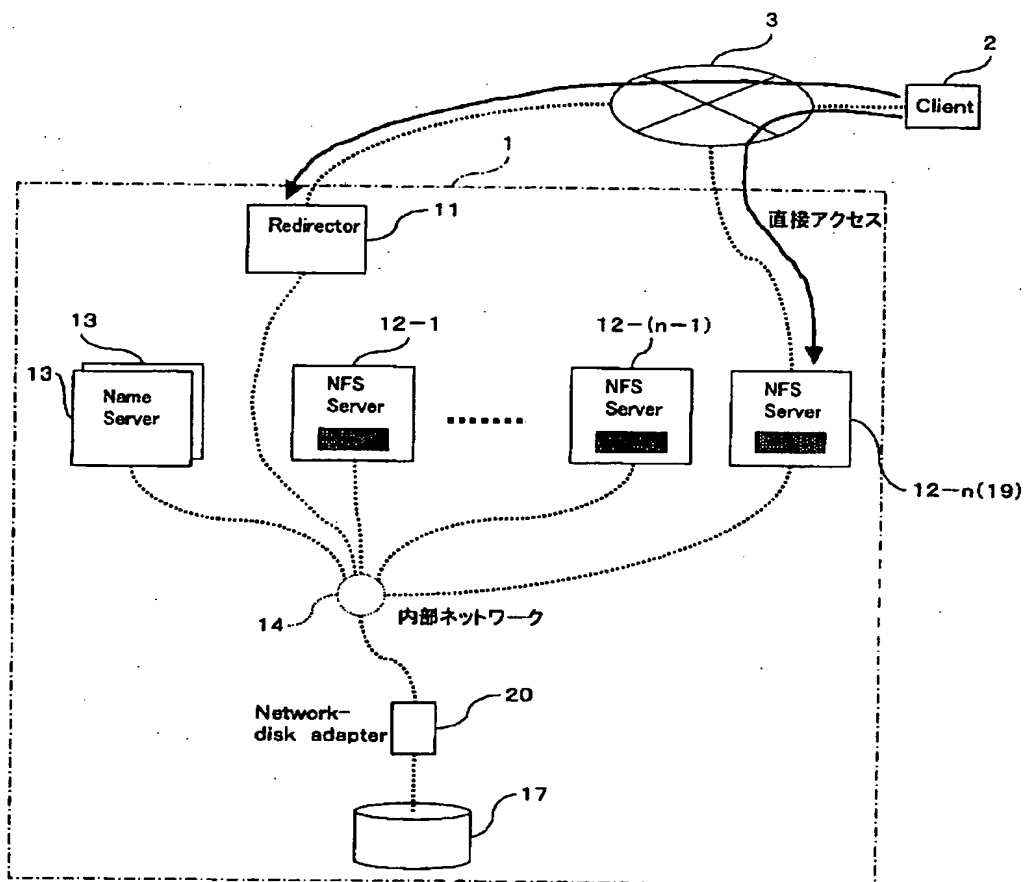
【図13】



【図14】



【図15】



PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-051890

(43)Date of publication of application : 23.02.2001

(51)Int.Cl.

G06F 12/00
G06F 13/00

(21)Application number : 11-226494

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 10.08.1999

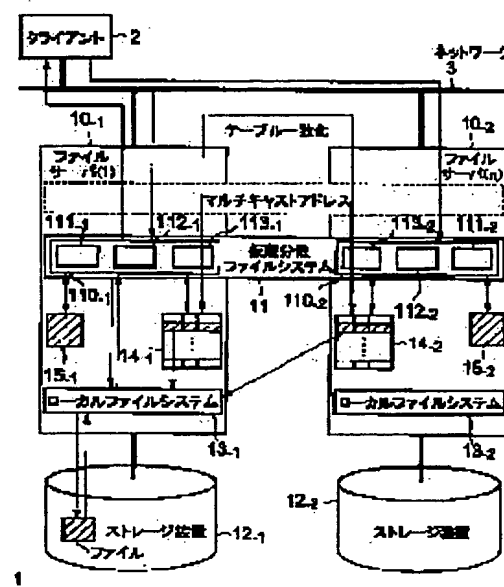
(72)Inventor : UCHIBORI IKUO
TAKAKUWA MASAYUKI

(54) VIRTUAL DECENTRALIZED FILE SERVER SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To make a client not pay attention to the number of file servers decentralized in a network and the connection states of storage devices.

SOLUTION: This virtual decentralized file server system 1 is equipped with servers 10-1 and 10-2 decentralized in the network 3 and a virtual decentralized file system 11 is decentralized and mounted on each of the servers. Modules 110-1 and 110-2 on the servers 10-1 and 10-2 which constitute this system 11 when receiving a file operation request multicast from a client 2 judge whether or not their servers are optimum servers capable of handling the request according to server information holding parts 15-1 and 15-2 holding mapping tables 14-1 and 14-2 between the virtual decentralized file system 11 and all local file systems 13-1 and 13-2 or server information on all the servers, and makes a local file system of a corresponding server perform requested file operation according to the judgement result.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. *** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] It has the following, the aforementioned virtual distributed file system. It consists of management modules formed in each aforementioned file server, respectively, each aforementioned management module. By receiving in common the file manipulation demand multicasted from the client, and referring to the aforementioned mapping table of a self-server according to the demand concerned, it judges whether it is the optimal server to which a self-server can process the demand concerned. The virtual distributed file server system characterized by being constituted so that the aforementioned corresponding local file system of a server may be made to perform demanded file manipulation, only when it is judged that it is the optimal server. The virtual distributed file system of not being dependent on the actual storage composition which is the virtual distributed file server system equipped with two or more file servers distributed on the network which can be multicasted, is distributed and mounted in each aforementioned file server, and manages the file of all file servers in integration. The local file system which is mounted independently in each aforementioned file server, respectively, and manages storage composition peculiar to each server. The mapping table which holds the information on mapping between the virtual distributed file server system concerned and the aforementioned local file system which actually manages the file about each file which is prepared in each aforementioned file server, respectively, and is managed in integration by the aforementioned virtual distributed file system.

[Claim 2] The virtual distributed file server system equipped with two or more file servers distributed on the network which can be multicasted characterized by providing the following. The virtual distributed file system of not being dependent on the actual storage composition which is distributed and mounted in each aforementioned file server, and manages the file of all file servers in integration. The local file system which is mounted independently in each aforementioned file server, respectively, and manages storage composition peculiar to each server. The mapping table which holds the information on mapping between the virtual distributed file server system concerned and the aforementioned local file system which actually manages the file about each file which is prepared in each aforementioned file server, respectively, and is managed in integration by the aforementioned virtual distributed file system. Either [at least] the information which is prepared in each aforementioned file server, respectively, and shows the availability of the storage equipment of the server about all the aforementioned file servers, or the information which shows the load situation of the server.

[Claim 3] The aforementioned management module is the virtual distributed file server system according to claim 1 or 2. It carries out [whether when the aforementioned file manipulation demand is a file read-out demand or a file write request, with reference to the aforementioned mapping table of a self-server, the corresponding file is under management of the aforementioned local file system of a self-server, and] whether it is the optimal server to which a self-server can process the aforementioned demand, and that it judges as the feature.

[Claim 4] It is the virtual distributed file-server system according to claim 2 characterized by to judge whether the aforementioned management module is comparing the availability of the storage equipment of the server, or the load situation of the server about each of all the

aforementioned servers with reference to the aforementioned server information maintenance means of a self-server, and is the optimal server to which a self-server can process the aforementioned demand when the aforementioned file manipulation demand is a new creation demand of a file.

[Claim 5] The aforementioned management module is a virtual distributed file server system according to claim 1 characterized by exchanging the information on the aforementioned mapping table of a self-server, and the information on the aforementioned mapping table of other servers by communication between servers in order to carry out identification of the content of the aforementioned mapping table of all the aforementioned file servers.

[Claim 6] In order that the aforementioned management module may carry out identification of the content of the aforementioned mapping table of all the aforementioned file servers. While exchanging the information on the aforementioned mapping table of a self-server, and the information on the aforementioned mapping table of other servers by communication between servers, in order to carry out identification of the content of the aforementioned server information maintenance means of all the aforementioned file servers. The virtual distributed file server system according to claim 2 characterized by exchanging the information on the aforementioned server information maintenance means of a self-server, and the information on the aforementioned server information maintenance means of other servers by communication between servers.

[Claim 7] It is prepared in each aforementioned file server, respectively, and a load status information maintenance means classified by file to hold the information which shows the load situation according to each file under management of the server is provided further. The aforementioned management module detects the file of the load which exceeded the 1st threshold from the information currently held at the aforementioned load status information maintenance means classified by file of a self-server. Communication between servers performs the replication of the file concerned to other arbitrary file servers. The virtual distributed file server system according to claim 1 or 2 characterized by leaving the processing to the demand concerned to a replication side when there is a read-out demand of the file concerned multicasted from the client.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001] [The technical field to which invention belongs] this invention carries out cooperation operation of two or more file servers which started the file server system in a computer network system, especially were connected on the network, and relates to the virtual distribution file server system operated as a single server from a client.

[0002] [Description of the Prior Art] Generally in today's computer network system, sharing a file between different computers connected to the network is performed. Under such environment, large-scale storage is connected to a specific computer, it applies as a file server or, recently, the system configuration of connecting the file server special-purpose machinery called NAS (Network Attached Storage) is taken in many cases.

[0003] In the environment (file server system) which uses a file server, if expandability is in a server side physically and efficiently when the storage capacity of a server runs short, it can be coped with by newly extending a disk unit etc. (storage equipment). At this time, it becomes the gestalt of using it, mounting new volume, from a client. Moreover, the server itself will be extended if the expandability of a server has reached the limitation. At this time, it becomes the gestalt of using it, mounting new volume after he is conscious of the extended server from a client.

[0004] [Problem(s) to be Solved by the Invention] When performing file sharing in the above-mentioned computer network system using a file server, it is common that the volume composition by the side of a file server is in sight as it is from a client side. For example, when a disk unit is extended by the server side, when new volume has been recognized, you have to mount a client side. Or when the server itself is extended, the complicated work of determination or system construction, management, etc. generates the employment policy of the extended server, and the new server has been recognized also by the client side, you have to mount new volume.

[0005] Thus, in the file-sharing system (file server system) using the conventional file server, when extension of a disk unit (storage equipment) or extension of a server was required, the problem that cost great for new setup and management occurred was in all of a client side the server side. Furthermore, there was a use gestalt of storage, when only capacity wanted to extend a specific file system as it is, and it also had the case which is not solved only by extending storage equipment and a server.

[0006] this invention was made in consideration of the above-mentioned situation, and the purpose can treat from a client two or more file servers distributed on the network as a single server, and is to offer the virtual distribution file server system which does not make a client conscious of the connection state of the number of a server, or storage equipment.

[0007] [Means for Solving the Problem] The virtual distributed file system of not being dependent on the actual storage composition which this invention is distributed and mounted in two or more file servers connected to the network which can be multicasted, and manages the file of all file

servers in integration. The local file system which is mounted independently in each file server, respectively, and manages storage composition peculiar to each server, it is prepared in each aforementioned file server, respectively, about each above-mentioned file information on mapping between a virtual distribution file server system and the local file system which actually manages the file (for example, it is managed by the virtual distribution file server system, and with the imagination path which appears from a client) While having a mapping table holding the information which matched the physical whereabouts which is managed with a local file system and is not visible from a client It is the management module in which the above-mentioned virtual distributed file system was prepared by each file server, respectively. By receiving in common the file manipulation demand multicasted from the client, and referring to the mapping table of a self-server according to the demand concerned Only when a self-server judges whether it is the optimal server which can process the demand concerned and judges that it is the optimal server, it is characterized by constituting with the management module to which the demanded file manipulation is made to perform with the corresponding local file system of a server.

[0008] The information which shows the availability of the storage equipment of the server about all file servers on each file server here, A server information maintenance means to hold the server information containing at least one side of the information which shows the load situation of the server is established further, and by each above-mentioned management module When the file manipulation demand multicasted from the client is received, it is good also as demand concerned by referring to the mapping table of a self-server, or a server information maintenance means according to the demand concerned.

[0009] In such composition, the file manipulation demand multicasted without being conscious of a specific file server from a client is received in common by the management module on each file server which constitutes a virtual distribution file server system, and the mapping table of a server (self-server) which corresponds according to the demand, or a server information maintenance means is referred to. And only the only server (upper management module) which it was judged whether it is the optimal server to which a self-server can process the above-mentioned demand, and was judged to be the optimal server makes the local file system of a self-server perform demanded file manipulation as a result of this reference.

[0010] Thus, from the client of a requiring agency, he can treat two or more file servers distributed on the network as a single server, and does not need to be conscious of the connection state of the number of a server, or storage equipment.

[0011] It is good to apply either the following 1st or the 4th algorithm (the judgment technique) as an algorithm for judging whether a self-server is optimal server by the above-mentioned management module here.

[0012] The 1st algorithm is technique judged by whether the file which is applied when a file manipulation demand is a file read-out demand or a file write request, and corresponds based on the information on the mapping table of a self-server is under management of the local file system of a self-server.

[0013] The 2nd algorithm is what is applied when a file manipulation demand is a new creation demand of a file. It is based on the information on the server information maintenance means of a self-server, about each of all servers It is the technique (for example, when the availability of a self-server is the largest, the load of a self-server judges it as the above-mentioned optimal server to a low case most) judged by comparing the availability (empty storage capacity) of the storage equipment of the server, or the load situation of the server.

[0014] The 3rd algorithm is also the technique (for example, when the size of a continuation field securable on the storage equipment of a self-server is the largest, it judges as the above-mentioned optimal server) which judges by being applied when a file-manipulation demand is a new creation demand of a file, asking for a continuation field securable on the storage equipment which corresponds about each of all servers based on the information on the mapping table of a self-server, and comparing the size of the continuation field.

[0015] The 4th algorithm is also the technique judged by being applied when a file manipulation

demand is a new creation demand of a file, calculating at least two, the availability of the storage equipment of the server, the load of the server, and a continuation field securable on the storage equipment concerned, about each of all servers, and comparing the at least two information searched for as compound condition.

[0016] Even if each server receives in common the file manipulation demand multicasted without being conscious of a specific file server from a client by applying any one of the 1st of a more than, or the 4th algorithm, it can judge autonomously whether it is the optimal server for performing the demanded file manipulation for the server itself, without communicating mutually each time.

[0017] It is good to give the function (communication module) to exchange the information on the mapping table of a self-server and the information on the mapping table of other servers by communication between servers to each above-mentioned management module here, in order to carry out identification of the content of the mapping table of all file servers. Moreover, it is good to give the function to exchange the information on the server information maintenance means of a self-server and the information on the server information maintenance means of other servers by communication between servers, further to each management module (inner communication module) with the composition which was equipped with the server information maintenance means on each server in addition to the mapping table, in order to carry out identification of the content of the server information maintenance means of all file servers. [0018] Moreover, for the identification of a mapping table, when the file organization actually managed with the local file system of a self-server is changed, it is efficient to transmit the changed information (mapping information) to all other servers by communication between servers. Similarly, for the identification of the content of a server information maintenance means, it is efficient to update the server information on a self-server periodically, and to transmit the updated server information to all other servers by communication between servers each time.

[0019] Moreover, this invention is characterized also by performing the 1st of the following [the management module of the server], or 4th processing, when a server is dynamically extended by the above-mentioned virtual distribution file server system. In the 1st processing, a lock setup which forbids renewal of a mapping table and a server information maintenance means to all other servers by communication between servers is performed. first, in the following processing of the 2nd The content of a mapping table server information maintenance means is copied to a self-server from other arbitrary servers by communication between servers, in the following processing of the 3rd The server information on a self-server is added to the server information maintenance means of a self-server. in the following processing of the 4th - BA information on a self-server is made to reflect in the server information maintenance means of all other servers by communication between servers, identification of the server information maintenance means of all servers is attained, and the above-mentioned lock setup is canceled after an appropriate time.

[0020] The number of a server is dynamically extensible with a series of operation at the time of such server extension. And a client can use the extended server, without being conscious of extension of the number of a server.

[0021] Moreover, this invention adds a load status information maintenance means classified by file to hold the information which shows the load situation according to each file under management of the server to each file server, and sets it to the management module of each server. The file of the load which exceeded the 1st threshold from the information currently held at the load status information maintenance means classified by file of a self-server is detected. Communication between servers performs the replication of the file concerned to other arbitrary file servers. When there is a read-out demand of the file concerned multicasted from the client, it is characterized also by leaving the processing to the demand concerned to a replication side. [0022] In such composition, an autonomous load distribution becomes possible. Here, an autonomous load distribution can carry out to a large area more effectively by communication between servers performing the replication of the file concerned to other arbitrary file servers, when the file which carried out [above-mentioned] detection is a file by which the replication

was carried out, and leaving the processing to the demand concerned to a new replication side, when there is a read-out demand of the file concerned multicasted from the client. Moreover, when the load of a file set as the object of the replication to other servers consists of the 1st threshold of the above below the threshold of a low 2nd Elimination of the file which corresponds by communication between servers to the other servers concerned from the management module on the server which performed the replication is required. When there is a read-out demand of the file concerned multicasted from the client, and self performs processing to the demand concerned, a dynamic load distribution becomes possible.

[0023] Now, it is possible the communication between servers for the above-mentioned identification (communication of mapping information or server information) and to use the above-mentioned network for communication between servers for the above-mentioned replication further. However, it is good to consider as the composition which forms the channel (private channel) of the exclusive use which interconnects each file server independently of the above-mentioned network, and performs communication between servers using the channel concerned. In this case, it can prevent that the throughput of a network gets worse for communication between servers.

[0024] Moreover, this invention is characterized also by performing the above-mentioned communication between servers by each above-mentioned management module through the interface concerned while it is further equipped with the interface in which the multi-host who interconnects the storage equipment of each file server and the server concerned is possible.

[0025] In such composition, since each above-mentioned storage equipment is shared between each server with the above-mentioned interface and communication between servers for the replication of the communication between servers for the above-mentioned identification and a still more dynamic file is performed through the above-mentioned interface, an autonomous load distribution is realized effectively.

[0026] In addition, this invention is materialized also as invention concerning a method.

[0027]

[Embodiments of the Invention] Hereafter, with reference to a drawing, it explains per gestalt of operation of this invention.

[0028] [Operation gestalt of ** 1st] drawing 1 is the block diagram showing the composition of the computer network system which applies the virtual distribution file server system concerning the 1st operation gestalt of this invention.

[0029] In this drawing, it is the client (client computer) as which 1 requires a virtual distribution file server system of this file server system 1, and 2 requires file service of it. The virtual distribution file server system 1 is realized using the plurality (two sets of for example, file servers) (server computer) 10-1 distributed on the network 3, and 10-2. In addition, drawing, although only one set is shown for convenience, as for a client 2, it is common that more than one exist.

[0030] 11 is a virtual distributed file system which makes the center of the virtual distribution file server system 1, and is distributed and mounted in each file server 10-1 and 10-2. this virtual distributed file system 11 -- all the file servers 10-1 and the file of 10-2 -- integration ---like - managing -- a file server 10-1 and 10-2 -- each actual volume composition (storage composition) is provided with the imagination file system for which it does not depend to a client 2

[0031] The virtual distributed file system 11 has each file server 10-1 and the virtual distribution file module 110-1, 110-2 distributed and mounted in 10-2. The virtual distribution file module 110-1, 110-2 is a management module for showing as one file system virtually to a client 2. distributing and processing the demand from a client 2 on a file server 10-1 and 10-2. The virtual distribution file module 110-1, 110-2 The virtual distribution file interface 111-1, 111-2 which processes nothing and the demand from a client 2 for the center of the module 110-1, 110-2 concerned. The local file system 13-1 mentioned later and the interface 112-1, 112-2 of 13-2 (local file interface). It has the communication module 113-1, 113-2 which performs communication (communication between servers represented by the information on 14-2, and server identification of the mapping table 14-1 mentioned later, the information on 14-2, and server

information) between the module 110-1,110-2 concerned.

[0032] A file server 10-1 and 10-2 are connected with the client 2 through the network 3. A file server 10-1 and 10-2 mount the storage equipments 12-1, such as a disk unit connected to the server 10-1 concerned and 10-2 other than the virtual distributed file system 11, the local file system (local file system) 13-1 which manages 12-2 (actual storage composition), and 13-2, respectively.

[0033] The virtual distributed file system 11, the local file system 13-1, the mapping table 14-1 of the same content which matches 13-2, and 14-2 are prepared in a file server 10-1 and 10-2. The data structure of this table 14-1 (1 = 2) is shown in drawing 2.

[0034] The registration field 141 of the file name of a file where the virtual distributed file system 1 manages each entry of table 14-1 (file name field). The logical whereabouts on the virtual distributed file system 1 of the file concerned Express. A path (it is visible from a client 2) The registration field of a (virtual path) (Virtual path field) 142, the registration field (whereabouts positional information field) 143 of the whereabouts positional information showing the physical whereabouts position on the storage of the file concerned (it is not visible from a client 2), and the access privilege (permission/prohibition) to the file concerned It has the registration field (permission information field) 144 of the permission information for managing, and the registration field 145 of other various attributes.

[0035] The virtual distributed file system 11 (upper virtual distribution file module 110-i) refers to mapping table 14-i of such a data structure -- for example, in a file server 10-1 and any of 10-2 a certain file's being and whereabouts information can be acquired, and also a permission etc. can acquire the attribute of a file if needed

[0036] The server information attaching part 15-1 of the same content and 15-2 are further prepared in a file server 10-1 and 10-2. Server information attaching part 15-i (1 = 2) is used for holding server information including all the file servers 10-1 that constitute the virtual distributed file server system 1, the information (resource information) which shows the empty storage capacity (storage equipment 12-1 and 12-2) of 10-2, and the information which shows a load situation as shown in drawing 3.

[0037] Next, operation of the composition of drawing 1 is explained. With this operation gestalt, it seems that each file server 10-1, the local file system 13-1 of 10-2, and not 13-2 but the virtual distributed file system 11 is mounted from the client 2. Then, a client 2 publishes the same demand to all the file servers 10-1 in which the virtual distributed file system 11 is mounted, and 10-2, when a certain file manipulation demand occurs. According to the technique of using IP (Internet Protocol) multicasting in this case, issue of a demand is possible for a client 2 side, without being conscious of the number of a file server.

[0038] A file server 10-1 and 10-2 will pass the demand concerned to the virtual distribution file module 110-1,110-2 corresponding to the self-server in the virtual distributed file system 11, if the demand from a client 2 is received. Then, a module 110-1,110-2 (inner virtual distribution file interface 111-1,111-2) is the read-out demand or the creation demand which it writes in (updating), and is a demand or is a creation demand or directory of a new file the demand of whose is a file, or distinguishes the demand classification.

[0039] Here, the file manipulation demand from a client 2 shall be a read-out demand or write request of a file. The file name of the file set as the object of a demand and the path on the virtual distributed file system 11 of the file concerned (virtual path) are given to this demand.

[0040] The virtual distribution file module 110-1,110-2 (inner virtual distribution file interface 111-1,111-2) When the file manipulation demand from a client 2 is a read-out demand or write request of a file. The file name and virtual path of a file which were demanded refer the mapping table 14-1 in a self-server, and 14-2. It investigates whether the file with the operation demand is held in the self-server (the storage equipment 12-1 connected to the self-server, 12-2) from the registration information on the table 14-1 with a file name and a virtual path concerned, and the whereabouts positional information field 143 in the entry in 14-2.

[0041] When the demanded file is held in the self-server, the virtual distribution file module 110-1,110-2 (inner virtual distribution file interface 111-1,111-2) accesses an actual file through the local file system 13-1 in a self-server, and 13-2 with the local file interface 112-1,112-2, and

returns a response to a client 2. On the other hand, when there is no file with the operation demand into a self-server, it is regarded as what other servers answer, and does not answer. [0042] On the other hand, when the demand from a client 2 is creation of a new file, or creation of a directory, refer to the server (not being mapping table 14-1 and 14-2) information attaching part 15-1 and 15-2 for the virtual distribution file module 110-1,110-2. And according to a predetermined algorithm, only virtual distribution file module 110-i on [of any one] server 10-i (i is 1 or 2) receives the demand from a client 2 by virtual distribution file interface 111-i based on the server information on the server information attaching part 15-1 and all the servers currently held 15-2. Specifically, virtual distribution file module 110-i on server 10-i shall receive the demand from a client 2, when the empty storage capacity which the server information on all servers shows is measured and it can be judged with the empty storage capacity of self-server 10-i (storage equipment 12-i) being the largest. In this case, it is not necessary to necessarily give the information on a load situation into server information.

[0043] In addition, the load which the server information on all servers shows is compared, and you may make it the load of a self-server receive the demand from a client 2 to a low case most. In this case, it is not necessary to necessarily give the information on the empty storage capacity of a server [be / under / server information / correspondence / it].

[0044] In addition, it asks for each storage equipment 12-1 and a continuation field securable on 12-2 from the mapping information for every file of mapping table 14-i (from the operating condition of jamming each storage equipment 12-1 and the field of 12-2), and when storage equipment with the largest size is storage equipment 12-i of a self-server, you may make it for the continuation field more than required size to be securable, and receive the demand from a client 2. In this case, the server information attaching part 15-1 and 15-2 are not necessarily required.

[0045] Furthermore, it is good as for a method of judgment in whether it is the optimal server to which is asked for an evaluation value by making at least two of the sizes of empty storage capacity, a load situation, and a securable continuation field into conditions (compound condition), and a self-server receives a demand.

[0046] Now, in virtual distribution file module 110-i on file server 10-i, if the demand from a client 2 is received by virtual distribution file interface 111-i, local file system 13-i will perform creation of the demanded new file, or creation of a directory through local file interface 112-i, and the mapping table 14-1 and the entry information applicable to 14-2 will be registered.

[0047] After creation of a new file or creation of a directory is completed, in virtual distribution file module 110-i on file server 10-i, the new entry information registered into mapping table 14-i on a self-server is sent to virtual distribution file module 110-j on all other server 10-j (j is 1 or 2, however j=i) through a network 3 by communication-module 113-i. Virtual distribution file module 110-j (inner virtual distribution file interface 111-j) receives the entry information on mapping table 14-i sent from virtual distribution file module 110-i through communication-module 113-j. And module 110-j (inner interface 111-j) received, and also registers the entry information of mapping table 14-i on server 10-i into mapping table 14-j in a self-server. Thus, the virtual distribution file module 110-1,110-2 on file server 10-1,110-2 can attain identification of the mapping table 14-1 concerned and the content of 14-2 by exchanging mutually the mapping table 14-1 and the entry information (entry information updated further) for which 14-2 was newly registered.

[0048] Moreover, a file server 10-1 and the virtual distribution file module 110-1,110-2 on 10-2 While updating periodically the server information on a self-server (empty storage capacity and load situation) among the server information attaching part 15-1 on a self-server, and the server information on each server currently held 15-2 By sending the updated server information to all other servers (upper virtual distribution file module 110-2,110-1) periodically through a network 3 (communication module 113-1,113-2) Identification of each file server 10-1, the server information attaching part 15-1 of 10-2, and the content of 15-2 is attained. That is, the virtual distribution file module 110-1,110-2 attains identification by exchanging server information mutually periodically.

[0049] By the above operation, distribution / cooperation operation of a file server 10-1 and 10-

2 can be carried out autonomously, and an imagination file server can be offered, without a file server making a client 2 in fact conscious of two sets (two or more sets) of a certain thing. [0050] In addition, although the example of the system of drawing 1 explained the case where the number of servers was two, even if a server is three or more sets, an imagination file server can be offered according to the same structure.

[0051] [Operation gestalt of ** 2nd] drawing 4 is the block diagram showing the composition of the computer network system which applies the virtual distribution file server system concerning the 2nd operation gestalt of this invention, and has given the same sign to the same portion as drawing 1.

[0052] In drawing 4, the virtual distributed file system 41 which makes the center of the virtual distribution file server system 4 is distributed and mounted in n sets of file servers 10-1 - , and 10-n, this virtual distributed file system 41 -- the virtual distributed file system 11 in drawing 1 - the same -- the file of all the file servers 10-1 - 10-n -- integration --like -- managing -- each file server 10-1 - 10-n -- each actual volume composition is provided with the imagination file system for which it does not depend to a client 2 The virtual distributed file system 41 has the virtual distribution file module 410-1 which processes the demand from a client 2 on each file server 10-1 - 10-n - 410-n. A module 410-1 - 410-n have the virtual distribution file interface 411-1 which is the same composition as the module 110-1, 110-2 in drawing 1 - 411-n, the local file interface 412-1 - 412-n, and a communication module 413-1 - 413-n. However, the communication module 413-1 in this operation gestalt - 413-n are constituted so that it may communicate through the private channel 5 mentioned later unlike the communication module 113-1, 113-2 in drawing 1.

[0053] A file server 10-1 - 10-n are connected with the client 2 through the network 3. A file server 10-1 - 10-n mount the local file system (local file system) 13-1 to 13-2 which manages the storage equipment 12-1 to 12-2 connected to the server 10-1 concerned - 10-n more than the virtual distributed file system 11, respectively. The mapping table 14-1 - 14-n, and the server information attaching part 15-1 to 15-2 are formed in a file server 10-1 - 10-n.

[0054] n sets of the point that the number of the file server from which the feature of the virtual distribution file server system 4 of the composition of drawing 4 constitutes a system unlike the virtual distribution file server system 1 of the composition of drawing 1 is n sets, and its file servers 10-1 - , and 10-n are the points by which interconnection is carried out also with private channel 5 with an another network 3. This private channel 5 is not specified about a physical layer, although it is Ethernet or a fiber channel (Fibre Channel). Moreover, you may be a loop and a switch although the bus type is assumed in the example of drawing 4 also about topology.

[0055] In the composition of drawing 4, in order for a file server 10-1 - 10-n to perform distribution / cooperation operation (the virtual distribution file module 410-1 in the virtual distributed file system 41 - 410-n) it is necessary to make the contents of the mapping table 14-1 - 14-n, and the server information attaching part 15-1 - 15-n always coincidence-ize among each server 10-1 - 10-n so that it may be guessed from explanation of operation with the operation form of the above 1st. However, when the number of the file server which constitutes the virtual distribution file server system 4 increases, the traffic (communication between servers of a sake) of the formation of information coincidence increases, and the throughput on a network 3 is made to get worse in performing information coincidence-ization between server 10-1 - 10-n through a network 3 like the operation form of the above 1st. [0056] Then, with this operation gestalt (2nd operation gestalt), the private channel 5 only for [between servers] information interchange is formed like the composition of drawing 4 among each file server 10-1 - 10-n. To the communication between servers performed by a communication module 413-1 - 413-n by the virtual distribution file module 410-1 in the virtual distributed file system 41 - 410-n That is, it is made to use this channel 5 for the communication between servers for carrying out identification of the content of the mapping table 14-1 - 14-n, and the server information attaching part 15-1 - 15-n.

[0057] Thus, with this operation gestalt, mitigation of the load of a network 3 can be aimed at by not being a network 3 and using the private channel 5 for the communication between servers for the identification of the content of the mapping table 14-1 - 14-n, and the server information

attaching part 15-1 - 15-n.

[0058] The 1st and 2nd operation gestalt described beyond [the 3rd operation gestalt] showed the example of the virtual distribution file server structure of a system which carries out distribution / cooperation operation of two or more file servers. The composition of this 1st [the 3. drawing 1 referred to with the 2nd operation gestalt, and drawing 4 is a static example in the specific number of a server. However, about the number of a server, considering as the composition which can be changed is desirable.

[0059] Then, the number of a server which constitutes a virtual distribution file server system is explained with reference to a drawing about the 3rd operation gestalt of this invention made dynamically extensible. Drawing 5 is the block diagram showing the composition of the computer network system which applies the virtual distribution file server system concerning the 3rd operation gestalt of this invention, and has given the same sign to the same portion as drawing 4.

[0060] First, as shown in drawing 5 (a), new file server 10- (n+1) shall be added to the virtual distribution file server system 4 4 shown in drawing 4, i.e., the virtual distribution file server system which consists of n sets of file servers 10-1 - , and 10-n.

[0061] In this case, virtual distribution file module 410- (n+1) on the virtual distributed file system 41 currently distributed by added file server 10- (n+1) To the file server 10-1 which already constitutes the virtual distribution file server system 4 - 10-n, as a sign A1 shows drawing 5 (a) (For example, the private channel which is not illustrated is minded) By communication between servers, renewal of the entry information on the mapping table 14-1 - 14-n and the server information on the server information attaching part 15-1 - 15-n (the resource information and load situation of each server are included) is locked.

[0062] Module 410- (n+1) on server 10- (n+1) moreover added As a sign A2 shows drawing 5 (b), from the server 10-1 of the either other file servers 10-1 or - the 10-n, for example, a file server All the information on the mapping table 14-1 and the server information attaching part 15-1 is copied to mapping table 14- in a self-server (n+1), and server information attaching part 15- (n+1) by communication between servers.

[0063] Next, to server information attaching part 15- after a copy (n+1), module 410- (n+1) on added file server 10- (n+1) adds the server information which shows the resource and load situation of a self-server, as a sign A3 shows drawing 5 (c).

[0064] After an appropriate time, as a sign A4 shows drawing 5 (d), module 410- (n+1) on file server 10- (n+1) publishes an identification demand of server information to all other file servers 10-1 - 10-n by communication between servers after an appropriate time, and cancels a lock after that.

[0065] By a series of above operation, a firewood shelf server (file server 10- (n+1)) can be dynamically added to the virtual distribution file server system 4 already built. If the information how a new resource is distributed to the volume composition of the present virtual distribution file server system 4 in this case is added, it is also possible to extend volume alternatively if needed.

[0066] [Operation gestalt of ** 4th] drawing 6 is the block diagram showing the composition of the computer network system which applies the virtual distribution file server system concerning the 4th operation gestalt of this invention, and has given the same sign to the same portion as drawing 4.

[0067] In drawing 6, 6 is a virtual distribution file server system equivalent to the virtual distribution file server system 4 in drawing 4. The feature of this virtual distribution file server system 6 is equipped with the load status information attaching part 16-1 classified by file holding the information (load status information classified by file) on the load situation about each file currently held at the self-server (storage equipment 12-1 - 12-n) - 16-n in the file server 10-1 which constitutes the system 6 concerned - 10-n. In connection with this, the functions which the virtual distributed file system 61 which makes the center of the virtual distribution file server system 6 has also differ in part in the virtual distributed file system 41 in drawing 4. However, the same sign (410-1 - 410-n) as drawing 4 is used for the virtual distribution file module of each file server 10-1 in the virtual distributed file system 61 - every

10-n for convenience. In addition, in drawing 6, the component in a module 410-1 - 410-n (a virtual distribution file interface, a local file interface, communication module) and the mapping table on a file server 10-1 - 10-n, and the server information attaching part are omitted.

[0068] Load status information attaching part 16 classified by file-i (i=1-n) has the data structure shown in drawing 7 (a), and holds the load status information classified by file containing the information which shows the load situation for every file in a self-server, the information (file attribute) which shows the attribute of the file, and a replication flag. It is shown whether the corresponding file of a file attribute is original or it is a replica (duplicate). Moreover, a replication flag shows whether it is whether it being generation ending about the replica at another servers side, and replication ending that is, when a corresponding flag is original.

[0069] Signs that the replica 612 of the arbitrary files 611 in the storage equipment 12-1 which a file server 10-1 has in storage equipment 12-n which file server 10-n has is held by the replication B1 are shown in drawing 6.

[0070] Next, operation of the composition of drawing 6 is explained. Refer to the load status information attaching part 16-1 classified by file - the 16-n for each virtual distribution file module 410-1 on the virtual distributed file system 61 - 410-n periodically, for example. And a module 410-1 - 410-n From the load status information according to file currently held at an attaching part 16-1 - 16-n In the file currently held at the self-server (storage equipment 12-1 - 12-n) When it is detected that the file of the load beyond the 1st threshold exists, communication between servers through the private channel 5 performs replication operation which generates asynchronously the replica of a file which corresponds to one of the other servers. The load situation of a file is total of the number of demands in the queue (queue) of the demand to the file concerned, or the size which the demand in the waiting state of the file concerned shows, and whenever it finishes processing a demand as whenever it receives a demand, it is updated here. Moreover, based on the server information currently held at the server information attaching part which is not illustrated, a load should just choose a low server as the object server of a replication most.

[0071] The virtual distribution file module 410-1 - 410-n will set to a notice state [finishing / a replication] the replication flag in the load status information of the corresponding file currently held at the load status information attaching part 16-1 classified by file of a self-server - 16-n, if replication operation is performed. Moreover, the virtual distribution file module of a server set as the object of replication operation adds the load status information of a replica [in the load status information attaching part classified by file of a self-server].

[0072] Here, as shown in drawing 6, the replication B1 of the file 611 which a file server 10-1 holds should be performed to file server 10-n through the private channel 5, and the replica 612 should be held at storage equipment 12- [of the file server 10-n concerned] n. In this case, the replication flag in the load status information of the file 611 currently held at the load status information attaching part 16-1 classified by file of a file server 10-1 is set to the state which shows replication ending. Moreover, the new load status information about the replica 612 of a file 611 is added to load status information attaching part 16 classified by file- [of file server 10-n] n. The file attribute in this load status information shows that a corresponding file is a replica (file 611) (612).

[0073] Henceforth, when there is a new read-out demand of a file 611 from a client 2, if the file server 10-1 (virtual distribution file module 410-1) holding the file 611 concerned investigated whether the load of the file 611 concerned would be over the 2nd threshold (however, the 2nd threshold < 1st threshold) and has exceeded, it will not answer a demand from a client 2. In this case, to the demand from a client 2, file server 10-n which received the replication answers. File server 10-n does not need to take into consideration whether a file server 10-1 answers, and as long as it has the replica 612 of the demanded file 611, it should just answer a client 2 here.

[0074] Thus, by file server 10-n processing the new read-out demand to the file 611 from a client 2 using the replica 612, by the file server 10-1 holding the file 611, processing of the read-out demand to the file 611 concerned received before it progresses, and the load of the file 611 concerned becomes below the threshold of the above 2nd. Then, the virtual distribution file module 410-1 on a file server 10-1 sends the demand for eliminating the replica 612 of a file 611

by communication between servers through the private channel 5 to virtual distribution file module 410-n on file server 10-n.

[0075] Virtual distribution file module 410-n on file server 10-n which received this demand processes using a replica 612 only to a received demand already, and eliminates the load status information corresponding to after an appropriate time with a replica 612. On the other hand, if the virtual distribution file module 410-1 on a file server 10-1 has the new read-out demand to the file 611 from a client 2, it will answer to it.

[0076] By the way, as a result of coming to receive the read-out demand to the file 611 concerned by file server 10-n before the load of the file 611 in a file server 10-1 becomes below the 2nd threshold by the replication of the file 611 from a file server 10-1 to file server 10-n, the load of the replica 612 of the file 611 concerned in file server 10-n can exceed the 1st threshold.

[0077] Then, what is necessary is for file server 10-n to generate the next generation's replica to other one server using a replica 612 shortly, namely, to perform the replication of a replication, and just to make the read-out demand to a file 611 process by the server in such a case, for that purpose, a load situation and a replication flag as shown in the load status information for every file held at load status information attaching part 16 classified by file-i (i=1-n) at drawing 7 (a) -- in addition, as shown in drawing 7 (b), it is good to give the generation information on a file

[0078] In this case, when the load of a certain generation's replica falls below in the threshold of the above 2nd, it is controllable to eliminate the replica of the next generation which the server set as the object of the replication by the server has from a server with the replica concerned etc. the case where the server as which elimination of a replica was required is generating the replica of the next generation further to another server at this time -- the -- it is good to eliminate the replica of the next generation further In addition, you may make it the server in which a load has a low file most answer the read-out demand to the file concerned about the same file (for a replica to be included) by attaining identification between each server at least like the server information described above about the load status information of the file relevant to the replica.

[0079] recently -- streaming data, such as video and an audio, -- or the contents of WWW (World Wide Web) etc. have fundamentally main read-out, its size is comparatively large, and the data which need a certain amount of response (it is a band guarantee depending on the case) are increasing them And since the case which looks at in the short term and access concentrates on specific data (file) is assumed, such data may be difficult to secure a response. The composition of drawing 8 described above is a thing supposing such a situation, and when access concentrates on a specific file, it enables it to distribute access of the FAIRUHE concerned by performing REBURKESHON of the file concerned automatically. Not only a load distribution but this composition can be used for backup of the high file of importance.

[0080] [Operation gestalt of ** 5th] drawing 8 is the block diagram showing the composition of the computer network system which applies the virtual distribution file server system concerning the 5th operation gestalt of this invention, and has given the same sign to the same portion as drawing 4.

[0081] In drawing 8, 8 is a virtual distribution file server system equivalent to the virtual distribution file server system 4 in drawing 4. The interconnection of a file server 10-1 - 10-n, and storage equipment 12-1 - 12-n is carried out by FC-AL (Fibre Channel Arbitrated Loop)80, and the feature of this virtual distribution file server system 8 is that it has applied the network configuration (that is, a multi-host is possible) which can share storage equipment (as a target) 12-1 - 12-n from each file server (as a host) 10-1 - 10-n. Here, unlike the composition of drawing 4, be careful of a point without the private channel 5.

[0082] What is necessary is just to perform communication between servers performed through the private channel 5 in the composition of drawing 4 (the communication module 413-1 of the virtual distribution file module 410-1 - 410-n - 413-n) through a network 3 with the composition of this drawing 8 like the composition of drawing 1 (as for drawing, this state is shown).

Moreover, you may be made to perform the above-mentioned communication between servers

on FC-AL80 through the interface for storage connection of a file server 10-1 - 10-n. In this case, the load of a network 3 is mitigable the same with having used the private channel 5.

[0083] According to the composition of drawing 8, since storage equipment 12-1 - 12-n can be directly seen from all the file servers 10-1 - 10-n, replication operation and a load distribution which were stated with the operation gestalt of the above 4th can be easily performed by giving the load status information attaching part 16-1 classified by file in drawing 6 - 16-n to each server 10-1 - 10-n. In addition, the network (interface) in which a multi-host is possible may not be restricted to FC-AL80, and may be a SCSI (Small Computer System Interface) bus.

[0084]

[Effect of the Invention] As explained in full detail above, according to this invention, from a client, he can treat two or more file servers distributed on the network as a single server, and a client is not made conscious of the connection state of the number of a server, or storage equipment.

[0085] Moreover, according to this invention, when a server is extended, volume can also be extended dynamically.

[0086] Furthermore, according to this invention, an autonomous load distribution is realizable among two or more servers.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. *** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] The block diagram showing the composition of the computer network system which applies the virtual distribution file server system concerning the 1st operation gestalt of this invention.

[Drawing 2] Drawing showing the example of a data structure of the mapping table in drawing 1.
[Drawing 3] Drawing showing the example of a data structure of the server information attaching part in drawing 1.

[Drawing 4] The block diagram showing the composition of the computer network system which applies the virtual distribution file server system concerning the 2nd operation gestalt of this invention.

[Drawing 5] The block diagram showing the composition of the computer network system which applies the virtual distribution file server system concerning the 3rd operation gestalt of this invention.

[Drawing 6] The block diagram showing the composition of the computer network system which applies the virtual distribution file server system concerning the 4th operation gestalt of this invention.

[Drawing 7] Drawing showing the example of a data structure of the load status information attaching part classified by file in drawing 6.

[Drawing 8] The timing chart explaining operation of this operation gestalt.

[Description of Notations]

1, 4, 6, 8 -- Virtual distribution file server system

2 -- Client

3 -- Network

5 -- Private channel

10-1 - 10-n -- File server

11, 41, 61 -- Virtual distributed file system

12-1 - 12-n -- Storage equipment

13-1 - 13-n -- Local file system

14-1 - 14-n -- Mapping table

15-1 - 15-n -- Server information attaching part

16-1 - 6-n -- Load status information attaching part classified by file

80 -- FC-AL (interface in which a multi-host is possible)

110-1 - 110-n, 410-1 - 410-n -- Virtual distribution file module (management module)

111-1 - 111-n -- Virtual distribution file interface

112-1 - 112-n -- Local file interface

113-1 - 113-n -- Communication module

611 -- File

612 -- (file 611) Replica

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2001-51890

(P2001-51890A)

(43) 公開日 平成13年2月23日 (2001.2.23)

(51) IntCl. ⁷	識別記号	F I	テーマコード* (参考)
G 0 6 F 12/00	5 4 5	G 0 6 F 12/00	5 4 5 A 5 B 0 8 2
13/00	3 5 1	13/00	3 5 1 E 5 B 0 8 9

審査請求 未請求 請求項の数7 O L (全 14 頁)

(21) 出願番号 特願平11-226494

(22) 出願日 平成11年8月10日 (1999.8.10)

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72) 発明者 内堀 郁夫

東京都府中市東芝町1番地 株式会社東芝

府中工場内

(72) 発明者 高桑 正幸

東京都府中市東芝町1番地 株式会社東芝

府中工場内

(74) 代理人 100058479

弁理士 鈴江 武彦 (外6名)

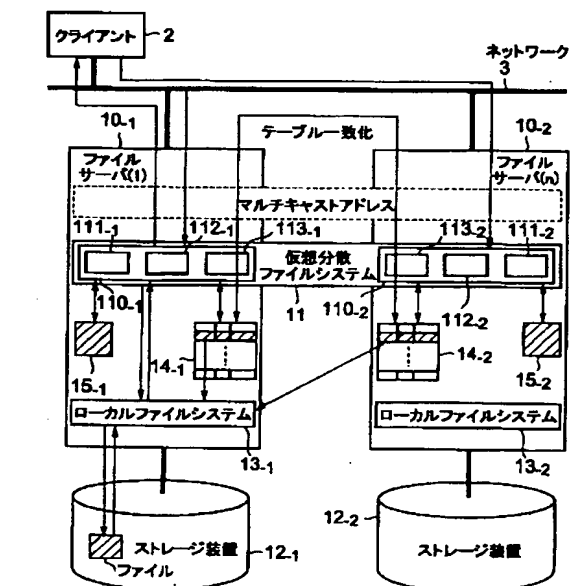
最終頁に続く

(54) 【発明の名称】 仮想分散ファイルサーバシステム

(57) 【要約】

【課題】 ネットワーク上に分散した複数のファイルサーバの台数やストレージ装置の接続状態をクライアントに意識させないで済むようにする。

【解決手段】 ネットワーク3上に分散したサーバ10-1, 10-2を備え、各サーバには、仮想分散ファイルシステム11が分散して実装されている。このシステム11を構成する、サーバ10-1, 10-2上のモジュール110-1, 110-2は、クライアント2からマルチキャストされたファイル操作要求を受け取ると、仮想分散ファイルシステム11と全ローカルファイルシステム13-1, 13-2とのマッピングテーブル14-1, 14-2または全サーバのサーバ情報を保持するサーバ情報保持部15-1, 15-2をもとに、自サーバが上記要求を処理可能な最適なサーバであるか否かを判断し、その判断結果に基づいて要求されたファイル操作に対応するサーバのローカルファイルシステムにより行わせる。



- 1
仮想分散ファイルサーバシステム
- 14-1, 14-2...マッピングテーブル
 - 15-1, 15-2...サーバ情報保持部
 - 110-1, 110-2...仮想分散ファイルモジュール
 - 111-1, 111-2...仮想分散ファイルインタフェース
 - 112-1, 112-2...ローカルファイルインタフェース
 - 113-1, 113-2...通信モジュール

【特許請求の範囲】

【請求項1】 マルチキャスト可能なネットワーク上に分散した複数のファイルサーバを備えた仮想分散ファイルサーバシステムであって、

前記各ファイルサーバに分散して実装され、全ファイルサーバのファイルを統合的に管理する、実際のストレージ構成には非依存の仮想分散ファイルシステムと、

前記各ファイルサーバにそれぞれ独立して実装され、各サーバに固有のストレージ構成を管理するローカルファイルシステムと、

前記各ファイルサーバにそれぞれ設けられ、前記仮想分散ファイルシステムで統合的に管理される各ファイルについて、当該仮想分散ファイルサーバシステムとそのファイルを実際に管理する前記ローカルファイルシステムとの間のマッピングの情報を保持するマッピングテーブルとを具備し、

前記仮想分散ファイルシステムは、前記各ファイルサーバにそれぞれ設けられた管理モジュールから構成され、前記各管理モジュールは、クライアントからマルチキャストされたファイル操作要求を共通に受け取り、当該要求に応じて自サーバの前記マッピングテーブルを参照することで、自サーバが当該要求を処理可能な最適なサーバであるか否かを判断し、最適なサーバであると判断した場合だけ、要求されたファイル操作を対応するサーバの前記ローカルファイルシステムにより行わせるように構成されていることを特徴とする仮想分散ファイルサーバシステム。

【請求項2】 マルチキャスト可能なネットワーク上に分散した複数のファイルサーバを備えた仮想分散ファイルサーバシステムであって、

前記各ファイルサーバに分散して実装され、全ファイルサーバのファイルを統合的に管理する、実際のストレージ構成には非依存の仮想分散ファイルシステムと、

前記各ファイルサーバにそれぞれ独立して実装され、各サーバに固有のストレージ構成を管理するローカルファイルシステムと、

前記各ファイルサーバにそれぞれ設けられ、前記仮想分散ファイルシステムで統合的に管理される各ファイルについて、当該仮想分散ファイルサーバシステムとそのファイルを実際に管理する前記ローカルファイルシステムとの間のマッピングの情報を保持するマッピングテーブルと、

前記各ファイルサーバにそれぞれ設けられ、全ての前記ファイルサーバについて、そのサーバのストレージ装置の空き容量を示す情報、及びそのサーバの負荷状況を示す情報の少なくとも一方を含むサーバ情報を保持するサーバ情報保持手段とを具備し、

前記仮想分散ファイルシステムは、前記各ファイルサーバにそれぞれ設けられた管理モジュールから構成され、前記各管理モジュールは、クライアントからマルチキャストされたファイル操作要求を共通に受け取り、当該要求に応じて自サーバの前記マッピングテーブルまたは前記サーバ情報保持手段を参照することで、自サーバが当該要求を処理可能な最適なサーバであるか否かを判断することを特徴とする請求項1または請求項2記載の仮想分散ファイルサーバシステム。

【請求項3】 前記管理モジュールは、前記ファイル操作要求がファイル読み出し要求またはファイル書き込み要求の場合には、自サーバの前記マッピングテーブルを参照し、該当するファイルが自サーバの前記ローカルファイルシステムの管理下にあるか否かにより、自サーバが前記要求を処理可能な最適なサーバであるか否かを判断することを特徴とする請求項1または請求項2記載の仮想分散ファイルサーバシステム。

【請求項4】 前記管理モジュールは、前記ファイル操作要求がファイルの新規作成要求の場合には、自サーバの前記サーバ情報保持手段を参照し、全ての前記サーバの各々について、そのサーバのストレージ装置の空き容量、またはそのサーバの負荷状況を比較することで、自サーバが前記要求を処理可能な最適なサーバであるか否かを判断することを特徴とする請求項2記載の仮想分散ファイルサーバシステム。

【請求項5】 前記管理モジュールは、全ての前記ファイルサーバの前記マッピングテーブルの内容を一致化するために、自サーバの前記マッピングテーブルの情報と他のサーバの前記マッピングテーブルの情報とをサーバ間通信により交換することを特徴とする請求項1記載の仮想分散ファイルサーバシステム。

【請求項6】 前記管理モジュールは、全ての前記ファイルサーバの前記マッピングテーブルの内容を一致化するために、自サーバの前記マッピングテーブルの情報と他のサーバの前記マッピングテーブルの情報とをサーバ間通信により交換する一方、全ての前記ファイルサーバの前記サーバ情報保持手段の内容を一致化するために、自サーバの前記サーバ情報保持手段の情報と他のサーバの前記サーバ情報保持手段の情報とをサーバ間通信により交換することを特徴とする請求項2記載の仮想分散ファイルサーバシステム。

【請求項7】 前記各ファイルサーバにそれぞれ設けられ、そのサーバの管理下にある各ファイル別の負荷状況を示す情報を保持するファイル別負荷状況情報保持手段を更に具備し、前記管理モジュールは、自サーバの前記ファイル別負荷状況情報保持手段に保持されている情報から第1の閾値を超えた負荷のファイルを検出して、他の任意のファイルサーバに対してサーバ間通信により当該ファイルのレプリケーションを行い、クライアントからマルチキャストされた当該ファイルの読み出し要求があった場合、当該要求に応じて自サーバの前記マッピングテーブルまたは前記サーバ情報保持手段を参照することで、自サーバが当該要求を処理可能な最適なサーバであるか否かを判断することを特徴とする請求項1または請求項2記載の仮想分散ファイルサーバシステム。

【請求項8】 前記各ファイルサーバにそれぞれ設けられ、そのサーバの管理下にある各ファイル別の負荷状況を示す情報を保持するファイル別負荷状況情報保持手段を更に具備し、前記管理モジュールは、自サーバの前記ファイル別負荷状況情報保持手段に保持されている情報から第1の閾値を超えた負荷のファイルを検出して、他の任意のファイルサーバに対してサーバ間通信により当該ファイルのレプリケーションを行い、クライアントからマルチキャストされた当該ファイルの読み出し要求があった場合、当該要求に応じて自サーバの前記マッピングテーブルまたは前記サーバ情報保持手段を参照することで、自サーバが当該要求を処理可能な最適なサーバであるか否かを判断することを特徴とする請求項1または請求項2記載の仮想分散ファイルサーバシステム。

【請求項9】 前記各ファイルサーバにそれぞれ設けられ、そのサーバの管理下にある各ファイル別の負荷状況を示す情報を保持するファイル別負荷状況情報保持手段を更に具備し、前記管理モジュールは、自サーバの前記ファイル別負荷状況情報保持手段に保持されている情報から第1の閾値を超えた負荷のファイルを検出して、他の任意のファイルサーバに対してサーバ間通信により当該ファイルのレプリケーションを行い、クライアントからマルチキャストされた当該ファイルの読み出し要求があった場合、当該要求に応じて自サーバの前記マッピングテーブルまたは前記サーバ情報保持手段を参照することで、自サーバが当該要求を処理可能な最適なサーバであるか否かを判断することを特徴とする請求項1または請求項2記載の仮想分散ファイルサーバシステム。

該要求に対する処理をレプリケーション側に任せるようにしたことを特徴とする請求項 1 または請求項 2 記載の仮想分散ファイルサーバシステム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、コンピュータ・ネットワークシステムにおけるファイルサーバシステムに係り、特にネットワーク上に接続された複数のファイルサーバを協調動作させて、クライアントからは単一のサーバとして機能させる仮想分散ファイルサーバシステムに関する。

【0002】

【従来の技術】今日のコンピュータ・ネットワークシステムにおいては、ネットワークに接続された異なるコンピュータ間でファイルを共有することが一般的に行われている。こうした環境下では、特定のコンピュータに大規模なストレージを接続して、ファイルサーバとして運用したり、最近ではNAS (Network Attached Storage) と呼ばれる、ファイルサーバ専用機を接続する等のシステム構成をとることが多い。

【0003】ファイルサーバを使用する環境（ファイルサーバシステム）では、サーバのストレージ容量が不足した場合には、サーバ側に物理的・性能的に拡張性があれば、新たにディスク装置等（のストレージ装置）を増設することで対処できる。このときクライアントからは、新たなボリュームをマウントして使用するという形態になる。また、サーバの拡張性が限界に達していれば、サーバ自体を増設することになる。このときクライアントからは、増設したサーバを意識した上で新たなボリュームをマウントして使用するという形態になる。

【0004】

【発明が解決しようとする課題】上記したコンピュータ・ネットワークシステムにおいてファイルサーバを利用してファイル共有を行う場合、クライアント側からは、ファイルサーバ側のボリューム構成がそのまま見えてしまうのが一般的である。例えばサーバ側でディスク装置を増設した場合には、クライアント側は新たなボリュームを認識した上で、マウントしなければならない。或いはサーバ自体を増設した場合には、増設したサーバの運用ポリシーを決定、もしくはシステム設定・管理等の煩雑な作業が発生する上、クライアント側でも、新たなサーバを認識した上で、新たなボリュームをマウントしなければならない。

【0005】このように従来のファイルサーバを用いたファイル共有システム（ファイルサーバシステム）では、ディスク装置（ストレージ装置）の増設、或いはサーバの増設が必要な場合、サーバ側、クライアント側のいずれにも、新たな設定・管理のために多大なコストが発生するという問題があった。更に、ストレージの利用形態によっては、特定のファイルシステムをそのまま容量

だけ拡張したい場合もあり、単にストレージ装置やサーバを増設するだけでは解決しないケースもあった。

【0006】本発明は上記事情を考慮してなされたものでその目的は、ネットワーク上に分散した複数のファイルサーバを、クライアントからは単一のサーバとして扱うことができ、サーバ台数やストレージ装置の接続状態をクライアントに意識させない仮想分散ファイルサーバシステムを提供することにある。

【0007】

【課題を解決するための手段】本発明は、マルチキャスト可能なネットワークに接続された複数のファイルサーバに分散して実装され、全ファイルサーバのファイルを統合的に管理する、実際のストレージ構成には非依存の仮想分散ファイルシステムと、各ファイルサーバにそれぞれ独立して実装され、各サーバに固有のストレージ構成を管理するローカルファイルシステムと、前記各ファイルサーバにそれぞれ設けられ、上記各ファイルについて、仮想分散ファイルサーバシステムとそのファイルを実際に管理するローカルファイルシステムとの間のマッピングの情報（例えば、仮想分散ファイルサーバシステムで管理され、クライアントから見える仮想的なバスと、ローカルファイルシステムで管理され、クライアントから見えない物理的な所在とを対応付けた情報）を保持するマッピングテーブルとを備えると共に、上記仮想分散ファイルシステムを、各ファイルサーバにそれぞれ設けられた管理モジュールであって、クライアントからマルチキャストされたファイル操作要求を共通に受け取り、当該要求に応じて自サーバのマッピングテーブルを参照することで、自サーバが当該要求を処理可能な最適なサーバであるか否かを判断し、最適なサーバであると判断した場合だけ、要求されたファイル操作を対応するサーバのローカルファイルシステムにより行わせる管理モジュールにより構成することを特徴とする。

【0008】ここで、各ファイルサーバ上に、全ファイルサーバについて、そのサーバのストレージ装置の空き容量を示す情報、及びそのサーバの負荷状況を示す情報の少なくとも一方を含むサーバ情報を保持するサーバ情報保持手段を更に設け、上記各管理モジュールでは、クライアントからマルチキャストされたファイル操作要求を受け取った場合に、当該要求に応じて自サーバのマッピングテーブルまたはサーバ情報保持手段を参照することで、自サーバが当該要求を処理可能な最適なサーバであるか否かを判断する構成としてもよい。

【0009】このような構成においては、クライアントから特定のファイルサーバを意識することなくマルチキャストされたファイル操作要求は、仮想分散ファイルサーバシステムを構成する各ファイルサーバ上の管理モジュールで共通に受け取られ、その要求に応じて対応するサーバ（自サーバ）のマッピングテーブルまたはサーバ情報保持手段が参照される。そして、この参照の結果、

自サーバが上記要求を処理可能な最適なサーバであるか否かが判断され、最適なサーバであると判断した唯一のサーバ（上の管理モジュール）だけが、要求されたファイル操作を自サーバのローカルファイルシステムにより行わせる。

【0010】このように、要求元のクライアントからは、ネットワーク上に分散した複数のファイルサーバを単一のサーバとして扱うことができ、サーバ台数やストレージ装置の接続状態を意識する必要がない。

【0011】ここで、上記管理モジュールで、自サーバが最適なサーバであるか否かを判断するためのアルゴリズムとして、以下の第1乃至第4のアルゴリズム（判断手法）のいずれかを適用するとよい。

【0012】第1のアルゴリズムは、ファイル操作要求がファイル読み出し要求またはファイル書き込み要求の場合に適用されるもので、自サーバのマッピングテーブルの情報に基づいて、該当するファイルが自サーバのローカルファイルシステムの管理下にあるか否かにより判断する手法である。

【0013】第2のアルゴリズムは、ファイル操作要求がファイルの新規作成要求の場合に適用されるもので、自サーバのサーバ情報保持手段の情報に基づいて、全てのサーバの各々について、そのサーバのストレージ装置の空き容量（空き記憶容量）、またはそのサーバの負荷状況を比較することで判断する（例えば、自サーバの空き容量が最も大きい場合、或いは自サーバの負荷が最も低い場合に上記最適サーバと判断する）手法である。

【0014】第3のアルゴリズムも、ファイル操作要求がファイルの新規作成要求の場合に適用されるもので、自サーバのマッピングテーブルの情報に基づいて、全てのサーバの各々について対応するストレージ装置上に確保可能な連続領域を求め、その連続領域のサイズを比較することで判断する（例えば、自サーバのストレージ装置上に確保可能な連続領域のサイズが最も大きい場合に上記最適サーバと判断する）手法である。

【0015】第4のアルゴリズムも、ファイル操作要求がファイルの新規作成要求の場合に適用されるもので、全てのサーバの各々について、そのサーバのストレージ装置の空き容量、そのサーバの負荷、及び当該ストレージ装置上に確保可能な連続領域の少なくとも2つを求め、その求めた少なくとも2つの情報を複合条件として比較することで判断する手法である。

【0016】以上の第1乃至第4のアルゴリズムのいずれか1つを適用することで、クライアントから特定のファイルサーバを意識することなくマルチキャストされたファイル操作要求を各サーバが共通に受け取っても、その要求されたファイル操作を行うのに最適なサーバであるか否かを、その都度相互に通信を行うことなく、そのサーバ自身で自律的に判断することができる。

【0017】ここで、上記各管理モジュールに、全ての

ファイルサーバのマッピングテーブルの内容を一致化するために、自サーバのマッピングテーブルの情報と他サーバのマッピングテーブルの情報とをサーバ間通信により交換する機能（通信モジュール）を持たせるとよい。また、マッピングテーブルに加えてサーバ情報保持手段を各サーバ上に備えた構成では、各管理モジュール（内の通信モジュール）に、全てのファイルサーバのサーバ情報保持手段の内容を一致化するために、自サーバのサーバ情報保持手段の情報と他のサーバのサーバ情報保持手段の情報とをサーバ間通信により交換する機能を更に変持せるとよい。

【0018】また、マッピングテーブルの一致化のためには、自サーバのローカルファイルシステムで実際に管理されるファイル構成が変更された場合に、その変更された情報（マッピング情報）をサーバ間通信により他の全サーバに送信するのが効率的である。同様に、サーバ情報保持手段の内容の一致化のためには、自サーバのサーバ情報を定期的に更新し、その都度、その更新されたサーバ情報をサーバ間通信により他の全サーバに送信するのが効率的である。

【0019】また本発明は、上記仮想分散ファイルサーバシステムにサーバが動的に増設された場合に、そのサーバの管理モジュールで以下の第1乃至第4の処理を行うようにしたことをも特徴とする。まず、第1の処理では、サーバ間通信により他の全てのサーバに対してマッピングテーブル及びサーバ情報保持手段の更新を禁止するロック設定を行い、次の第2の処理では、サーバ間通信により他の任意のサーバからマッピングテーブルサーバ情報保持手段の内容を自サーバにコピーし、次の第3の処理では、自サーバのサーバ情報保持手段に自サーバのサーバ情報を追加し、次の第4の処理では、サーバ間通信により自サーバのサーバ情報を他の全てのサーバのサーバ情報保持手段に反映させて全サーバのサーバ情報保持手段の一致化を図り、しかる後に上記ロック設定を解除する。

【0020】このようなサーバ増設時の一連の動作により、動的にサーバ台数を拡張できる。しかもクライアントは、サーバ台数の拡張を意識することなく、増設されたサーバを利用することができる。

【0021】また本発明は、各ファイルサーバに、そのサーバの管理下にある各ファイル別の負荷状況を示す情報を保持するファイル別負荷状況情報保持手段を付加し、各サーバの管理モジュールにおいて、自サーバのファイル別負荷状況情報保持手段に保持されている情報から第1の閾値を超えた負荷のファイルを検出して、他の任意のファイルサーバに対してサーバ間通信により当該ファイルのレプリケーションを行い、クライアントからマルチキャストされた当該ファイルの読み出し要求があった場合、当該要求に対する処理をレプリケーション側に任せるようにしたことをも特徴とする。

【0022】このような構成においては、自律的な負荷分散が可能となる。ここで、上記検出したファイルがレプリケーションされたファイルである場合にも、他の任意のファイルサーバに対してサーバ間通信により当該ファイルのレプリケーションを行い、クライアントからマルチキャストされた当該ファイルの読み出し要求があった場合に、当該要求に対する処理を新たなレプリケーション側に任せることにより、自律的な負荷分散がより広範囲に効果的に行える。また、他サーバへのレプリケーションの対象となったファイルの負荷が前記第1の閾値より低い第2の閾値以下となった場合に、そのレプリケーションを行ったサーバ上の管理モジュールから当該他サーバに対してサーバ間通信により対応するファイルの消去を要求し、クライアントからマルチキャストされた当該ファイルの読み出し要求があった場合、当該要求に対する処理を自身が行うことにより、動的な負荷分散が可能となる。

【0023】さて、上記一致化のためのサーバ間通信（マッピング情報またはサーバ情報の通信）、更には上記レプリケーションのためのサーバ間通信には、上記ネットワークを用いることが可能である。しかし、各ファイルサーバを相互接続する専用の通信路（プライベート通信路）を上記ネットワークから独立に設け、当該通信路を用いてサーバ間通信を行う構成とするとよい。この場合、サーバ間通信のためにネットワークのスループットが悪化するのを防止できる。

【0024】また本発明は、各ファイルサーバ及び当該サーバのストレージ装置を相互接続するマルチホストが可能なインタフェースを更に備えると共に、上記各管理モジュールによる上記サーバ間通信を当該インタフェースを介して行うようにしたことをも特徴とする。

【0025】このような構成においては、上記各ストレージ装置を上記インタフェースによって各サーバ間で共有し、上記一致化のためのサーバ間通信、更には動的なファイルのレプリケーションのためのサーバ間通信が上記インタフェースを通して行われるため、自律的な負荷分散が効果的に実現される。

【0026】なお、本発明は方法に係る発明としても成立する。

【0027】

【発明の実施の形態】以下、本発明の実施の形態につき図面を参照して説明する。

【0028】【第1の実施形態】図1は本発明の第1の実施形態に係る仮想分散ファイルサーバシステムを適用するコンピュータ・ネットワークシステムの構成を示すブロック図である。

【0029】同図において、1は仮想分散ファイルサーバシステム、2は同ファイルサーバシステム1にファイルサービスを要求するクライアント（クライアントコンピュータ）である。仮想分散ファイルサーバシステム1

は、ネットワーク3上に分散配置された複数、例えば2台のファイルサーバ（サーバコンピュータ）10-1、10-2を用いて実現される。なお、図ではクライアント2は便宜的に1台だけが示されているが、複数存在するのが一般的である。

【0030】11は仮想分散ファイルサーバシステム1の中心をなす仮想分散ファイルシステムであり、各ファイルサーバ10-1、10-2に分散して実装されている。この仮想分散ファイルシステム11は、全ファイルサーバ10-1、10-2のファイルを統合的に管理し、ファイルサーバ10-1、10-2それぞれの実際のボリューム構成（ストレージ構成）には依存しない、仮想的なファイルシステムをクライアント2に対して提供するものである。

【0031】仮想分散ファイルシステム11は、各ファイルサーバ10-1、10-2に分散して実装された仮想分散ファイルモジュール110-1、110-2を有している。仮想分散ファイルモジュール110-1、110-2は、ファイルサーバ10-1、10-2上でクライアント2からの要求を分散して処理しつつ、クライアント2に対しては仮想的に1つのファイルシステムとして見せるための管理モジュールである。仮想分散ファイルモジュール110-1、110-2は、当該モジュール110-1、110-2の中心をなし、クライアント2からの要求を処理する仮想分散ファイルインタフェース111-1、111-2と、後述するローカルファイルシステム13-1、13-2とのインタフェース（ローカルファイルインタフェース）112-1、112-2と、当該モジュール110-1、110-2間での通信（後述するマッピングテーブル14-1、14-2の情報、及びサーバ情報の一致化のための通信に代表されるサーバ間通信）を行う通信モジュール113-1、113-2を持つ。

【0032】ファイルサーバ10-1、10-2はネットワーク3を介してクライアント2と接続されている。ファイルサーバ10-1、10-2は、仮想分散ファイルシステム11の他に、それぞれ当該サーバ10-1、10-2に接続されたディスク装置などのストレージ装置12-1、12-2（実際のストレージ構成）を管理するローカルなファイルシステム（ローカルファイルシステム）13-1、13-2を実装している。

【0033】ファイルサーバ10-1、10-2には、仮想分散ファイルシステム11とローカルファイルシステム13-1、13-2とを対応付ける同一内容のマッピングテーブル14-1、14-2が設けられている。このテーブル14-i（i=1, 2）のデータ構造を図2に示す。

【0034】テーブル14-iの各エントリは、仮想分散ファイルシステム1が管理するファイルのファイル名の登録フィールド（ファイル名フィールド）141と、当該ファイルの仮想分散ファイルシステム1上の論理的な所在を表す（クライアント2から見える）パス（仮想パ

ス)の登録フィールド(仮想パスフィールド)142、当該ファイルのストレージ上の物理的な所在位置を表す(クライアント2から見えない)所在位置情報の登録フィールド(所在位置情報フィールド)143、当該ファイルへのアクセス権(許可/禁止)を管理するためのパーミッション情報の登録フィールド(パーミッション情報フィールド)144、及びその他の各種属性の登録フィールド145を有している。

【0035】仮想分散ファイルシステム11(上の仮想分散ファイルモジュール110-i)は、このようなデータ構造のマッピングテーブル14-iを参照することにより、例えばあるファイルがファイルサーバ10-1、10-2のいずれにあるか等の所在情報を得ることができる。他、パーミッション等、必要に応じてファイルの属性を得ることができる。

【0036】ファイルサーバ10-1、10-2には更に、同一内容のサーバ情報保持部15-1、15-2が設けられている。サーバ情報保持部15-i(i=1,2)は、図3に示すように、仮想分散ファイルサーバシステム1を構成する全てのファイルサーバ10-1、10-2の(ストレージ装置12-1、12-2の)空き記憶容量を示す情報(リソース情報)、及び負荷状況を示す情報を含むサーバ情報を保持するのに用いられる。

【0037】次に図1の構成の動作を説明する。本実施形態では、クライアント2からは、各ファイルサーバ10-1、10-2のローカルファイルシステム13-1、13-2ではなくて、仮想分散ファイルシステム11がマウントされているように見えている。そこでクライアント2は、何らかのファイル操作要求が発生した場合、仮想分散ファイルシステム11が実装されている全ファイルサーバ10-1、10-2に対して同一の要求を発行する。この場合、例えばIP(Internet Protocol)マルチキャストを使用する等の手法によれば、クライアント2側はファイルサーバの台数を意識することなく要求の発行が可能である。

【0038】ファイルサーバ10-1、10-2は、クライアント2からの要求を受け取ると、当該要求を仮想分散ファイルシステム11内の自サーバに対応した仮想分散ファイルモジュール110-1、110-2に渡す。すると、モジュール110-1、110-2(内の仮想分散ファイルインタフェース111-1、111-2)は、その要求がファイルの読み出し要求もしくは書き込み(更新)要求であるか、または新規ファイルの作成要求もしくはディレクトリの作成要求であるか、その要求種別を判別する。

【0039】ここで、クライアント2からのファイル操作要求がファイルの読み出し要求もしくは書き込み要求であるものとする。この要求には、要求の対象となるファイルのファイル名と、当該ファイルの仮想分散ファイルシステム11上のパス(仮想パス)が付されている。

【0040】仮想分散ファイルモジュール110-1、110-2(内の仮想分散ファイルインタフェース111-1、111-2)は、クライアント2からのファイル操作要求がファイルの読み出し要求もしくは書き込み要求の場合、要求されたファイルのファイル名及び仮想パスにより自サーバ内のマッピングテーブル14-1、14-2を参照し、当該ファイル名及び仮想パスを持つテーブル14-1、14-2内エントリ中の所在位置情報フィールド143の登録情報から、操作要求のあったファイルが自サーバ内(自サーバに接続されたストレージ装置12-1、12-2)に保持されているか否かを調べる。

【0041】もし、要求されたファイルが自サーバ内に保持されている場合には、仮想分散ファイルモジュール110-1、110-2(内の仮想分散ファイルインタフェース111-1、111-2)は、ローカルファイルインタフェース112-1、112-2により自サーバ内のローカルファイルシステム13-1、13-2を介して実際のファイルにアクセスし、クライアント2に応答を返す。一方、操作要求のあったファイルが自サーバ内になかった場合には、他のサーバが応答するものと見なして応答しない。

【0042】これに対し、クライアント2からの要求が新規ファイルの作成、或いはディレクトリの作成であった場合には、仮想分散ファイルモジュール110-1、110-2は、(マッピングテーブル14-1、14-2ではなくて)サーバ情報保持部15-1、15-2を参照する。そして、サーバ情報保持部15-1、15-2に保持されている全サーバのサーバ情報をもとに、所定のアルゴリズムに従い、いずれか1つのサーバ10-i(iは1または2)上の仮想分散ファイルモジュール110-iだけが、仮想分散ファイルインタフェース111-iによりクライアント2からの要求を受け付ける。具体的には、サーバ10-i上の仮想分散ファイルモジュール110-iは、全サーバのサーバ情報の示す空き記憶容量を比較し、自サーバ10-i(のストレージ装置12-i)の空き記憶容量が最も大きいと判定できる場合に、クライアント2からの要求を受け付けるものとする。この場合、必ずしもサーバ情報中に負荷状況の情報を持たせる必要はない。

【0043】なお、全サーバのサーバ情報の示す負荷を比較し、自サーバの負荷が最も低い場合にクライアント2からの要求を受け付けるようにしてもよい。この場合、必ずしもサーバ情報中に対応するサーバの空き記憶容量の情報を持たせる必要はない。

【0044】この他に、マッピングテーブル14-iの各ファイル毎のマッピング情報から(つまり各ストレージ装置12-1、12-2の領域の使用状況から)、各ストレージ装置12-1、12-2上に確保可能な連続領域を求め、必要なサイズ以上の連続領域が確保でき、且つそのサイズが最も大きいストレージ装置が自サーバのストレージ装置12-iの場合に、クライアント2からの要求を

受け付けるようにしてもよい。この場合、サーバ情報保持部15-1、15-2は必ずしも必要でない。

【0045】更に、空き記憶容量と負荷状況と確保できる連続領域のサイズの少なくとも2つを条件（複合条件）として評価値を求め、自サーバが要求を受け付ける最適なサーバであるか否かを判断するようにしてもよい。

【0046】さて、ファイルサーバ10-i上の仮想分散ファイルモジュール110-iでは、仮想分散ファイルインタフェース111-iによりクライアント2からの要求を受け付けると、要求された新規ファイルの作成、或いはディレクトリの作成を、ローカルファイルインタフェース112-iを介してローカルファイルシステム13-iにより行い、マッピングテーブル14-1、14-2に該当するエントリ情報を登録する。

【0047】新規ファイルの作成、或いはディレクトリの作成が完了した後は、ファイルサーバ10-i上の仮想分散ファイルモジュール110-iでは、自サーバ上のマッピングテーブル14-iに登録した新たなエントリ情報を通信モジュール113-iによりネットワーク3を介して他の全てのサーバ10-j（jは1または2、但しj≠i）上の仮想分散ファイルモジュール110-jに送る。仮想分散ファイルモジュール110-j（内の仮想分散ファイルインタフェース111-j）は、仮想分散ファイルモジュール110-iから送られたマッピングテーブル14-iのエントリ情報を通信モジュール113-jを介して受け取る。そしてモジュール110-j（内のインタフェース111-j）は、受け取った他サーバ10-iのマッピングテーブル14-iのエントリ情報を自サーバ内のマッピングテーブル14-jに登録する。このように、ファイルサーバ10-1、110-2上の仮想分散ファイルモジュール110-1、110-2が相互にマッピングテーブル14-1、14-2の新規登録されたエントリ情報（更には更新されたエントリ情報）を交換し合うことで、当該マッピングテーブル14-1、14-2の内容の一致化を図ることができる。

【0048】また、ファイルサーバ10-1、10-2上の仮想分散ファイルモジュール110-1、110-2は、自サーバ上のサーバ情報保持部15-1、15-2に保持されている各サーバのサーバ情報のうち、自サーバのサーバ情報（空き記憶容量、及び負荷状況）を定期的に更新すると共に、その更新したサーバ情報を（通信モジュール113-1、113-2により）ネットワーク3を介して他の全てのサーバ（上の仮想分散ファイルモジュール110-2、110-1）に定期的に送ることで、各ファイルサーバ10-1、10-2のサーバ情報保持部15-1、15-2の内容の一致化を図る。つまり仮想分散ファイルモジュール110-1、110-2は定期的にサーバ情報を交換し合うことで一致化を図る。

【0049】以上の動作によって、ファイルサーバ10-1、10-2を自律的に分散・協調動作させることがで

き、クライアント2には実際にはファイルサーバが2台（複数台）あることを意識させずに、仮想的なファイルサーバを提供することができる。

【0050】なお、図1のシステムの例ではサーバが2台である場合について説明したが、サーバが3台以上であっても同様の仕組みによって、仮想的なファイルサーバを提供することができる。

【0051】【第2の実施形態】図4は本発明の第2の実施形態に係る仮想分散ファイルサーバシステムを適用するコンピュータ・ネットワークシステムの構成を示すブロック図であり、図1と同一部分には同一符号を付してある。

【0052】図4において、仮想分散ファイルサーバシステム4の中心をなす仮想分散ファイルシステム41は、n台のファイルサーバ10-1～10-nに分散して実装されている。この仮想分散ファイルシステム41は、図1中の仮想分散ファイルシステム11と同様に、全ファイルサーバ10-1～10-nのファイルを統合的に管理し、各ファイルサーバ10-1～10-nそれぞれの実際のボリューム構成には依存しない、仮想的なファイルシステムをクライアント2に対して提供している。仮想分散ファイルシステム41は、各ファイルサーバ10-1～10-n上でクライアント2からの要求を処理する仮想分散ファイルモジュール410-1～410-nを有している。モジュール410-1～410-nは、図1中のモジュール110-1、110-2と同様の構成である、仮想分散ファイルインタフェース411-1～411-nと、ローカルファイルインタフェース412-1～412-nと、通信モジュール413-1～413-nとを持つ。但し、本実施形態における通信モジュール413-1～413-nは、図1中の通信モジュール113-1、113-2と異なり、後述するプライベート通信路5を介して通信を行うように構成されている。

【0053】ファイルサーバ10-1～10-nはネットワーク3を介してクライアント2と接続されている。ファイルサーバ10-1～10-nは、仮想分散ファイルシステム11の他に、それぞれ当該サーバ10-1～10-nに接続されたストレージ装置12-1～12-2を管理するローカルなファイルシステム（ローカルファイルシステム）13-1～13-2を実装している。ファイルサーバ10-1～10-nには、マッピングテーブル14-1～14-nと、サーバ情報保持部15-1～15-2とが設けられている。

【0054】図4の構成の仮想分散ファイルサーバシステム4の特徴は、図1の構成の仮想分散ファイルサーバシステム1と異なって、システムを構成するファイルサーバの台数がn台である点と、そのn台のファイルサーバ10-1～10-nがネットワーク3とは別のプライベート通信路5によっても相互接続されている点である。このプライベート通信路5は、例えばイーサネット、或いはファイバチャネル（Fibre Channel）等であるが、

物理層に関しては特定しない。またトポロジに関しても、図4の例ではバス型を想定しているが、ループやスイッチであってもよい。

【0055】図4の構成において、ファイルサーバ10-1~10-nが（仮想分散ファイルシステム41内の仮想分散ファイルモジュール410-1~410-nにより）分散・協調動作を行うためには、前記第1の実施形態での動作説明から類推されるように、マッピングテーブル14-1~14-n、及びサーバ情報保持部15-1~15-nの内容を、各サーバ10-1~10-n間で常に一致化させておく必要がある。しかし、サーバ10-1~10-n間の情報一致化を、前記第1の実施形態と同様にネットワーク3を介して行うのでは、仮想分散ファイルサーバシステム4を構成するファイルサーバの台数が増加した場合には、その情報一致化の（ためのサーバ間通信の）トラフィックが増加し、ネットワーク3上のスループットを悪化させることになる。

【0056】そこで本実施形態（第2の実施形態）では、図4の構成のように、各ファイルサーバ10-1~10-n間にサーバ間の情報交換専用のプライベート通信路5を設け、仮想分散ファイルシステム41内の仮想分散ファイルモジュール410-1~410-nで通信モジュール413-1~413-nにより行われるサーバ間通信に、即ちマッピングテーブル14-1~14-n、及びサーバ情報保持部15-1~15-nの内容を一致化するためのサーバ間通信に、この通信路5を使用するようにしている。

【0057】このように本実施形態では、マッピングテーブル14-1~14-n、及びサーバ情報保持部15-1~15-nの内容の一致化のためのサーバ間通信に、ネットワーク3でなくてプライベート通信路5を用いることにより、ネットワーク3の負荷の軽減を図ることができる。

【0058】〔第3の実施形態〕以上に述べた第1、第2の実施形態では、複数のファイルサーバを分散・協調動作させる仮想分散ファイルサーバシステムの構成例を示した。この第1、第2の実施形態で参照した図1、図4の構成は、特定のサーバ台数における静的な例である。しかし、サーバ台数については、変更可能な構成とすることが好ましい。

【0059】そこで、仮想分散ファイルサーバシステムを構成するサーバ台数を動的に拡張可能とした本発明の第3の実施形態について図面を参照して説明する。図5は本発明の第3の実施形態に係る仮想分散ファイルサーバシステムを適用するコンピュータ・ネットワークシステムの構成を示すブロック図であり、図4と同一部分には同一符号を付してある。

【0060】まず、図4に示した仮想分散ファイルサーバシステム4、即ちn台のファイルサーバ10-1~10-nで構成される仮想分散ファイルサーバシステム4に、図5（a）に示すように、新たなファイルサーバ10-

(n+1)を追加するものとする。

【0061】この場合、追加されたファイルサーバ10-(n+1)にも分散されている仮想分散ファイルシステム41上の仮想分散ファイルモジュール410-(n+1)は、既に仮想分散ファイルサーバシステム4を構成しているファイルサーバ10-1~10-nに対し、図5（a）において符号A1で示すように、（例えば図示せぬプライベート通信路を介しての）サーバ間通信により、マッピングテーブル14-1~14-nのエントリ情報及びサーバ情報保持部15-1~15-nのサーバ情報（各サーバのリソース情報及び負荷状況を含む）の更新をロックする。

【0062】その上で、追加されたサーバ10-(n+1)上のモジュール410-(n+1)は、他のファイルサーバ10-1~10-nのうちのいずれかのサーバ、例えばファイルサーバ10-1から、図5（b）において符号A2で示すように、マッピングテーブル14-1及びサーバ情報保持部15-1の全情報を、サーバ間通信により自サーバ内のマッピングテーブル14-(n+1)及びサーバ情報保持部15-(n+1)にコピーする。

【0063】次に、追加されたファイルサーバ10-(n+1)上のモジュール410-(n+1)は、コピー後のサーバ情報保持部15-(n+1)に対し、図5（c）において符号A3で示すように、自サーバのリソース及び負荷状況を示すサーバ情報を追加する。

【0064】しかる後にファイルサーバ10-(n+1)上のモジュール410-(n+1)は、図5（d）において符号A4で示すように、サーバ間通信により他の全ファイルサーバ10-1~10-nに対してサーバ情報の一致化要求を発行し、その後にロックを解除する。

【0065】以上の一連の動作により、既に構築されている仮想分散ファイルサーバシステム4に対して、動的に新たなサーバ（ファイルサーバ10-(n+1)）を追加することができる。この場合、例えば現在の仮想分散ファイルサーバシステム4のボリューム構成に対し、新規リソースをどのように振り分けるか、といった情報を付加すれば、必要に応じてボリュームを選択的に拡張することも可能である。

【0066】〔第4の実施形態〕図6は本発明の第4の実施形態に係る仮想分散ファイルサーバシステムを適用するコンピュータ・ネットワークシステムの構成を示すブロック図であり、図4と同一部分には同一符号を付してある。

【0067】図6において、6は図4中の仮想分散ファイルサーバシステム4に相当する仮想分散ファイルサーバシステムである。この仮想分散ファイルサーバシステム6の特徴は、当該システム6を構成するファイルサーバ10-1~10-n内に、自サーバ（のストレージ装置12-1~12-n）に保持されている各ファイルについての負荷状況の情報（ファイル別負荷状況情報）を保持するファイル別負荷状況情報保持部16-1~16-nを備えて

いる点にある。これに伴い、仮想分散ファイルサーバシステム6の中心をなす仮想分散ファイルシステム61の持つ機能も図4中の仮想分散ファイルシステム41とは一部異なる。但し、仮想分散ファイルシステム61内の各ファイルサーバ10-1~10-n毎の仮想分散ファイルモジュールには便宜的に図4と同一符号(410-1~410-n)を用いている。なお、図6では、モジュール410-1~410-n内の構成要素(仮想分散ファイルインタフェース、ローカルファイルインタフェース、通信モジュール)、及びファイルサーバ10-1~10-n上のマッピングテーブル、サーバ情報保持部は省略されている。

【0068】ファイル別負荷状況情報保持部16-i ($i=1\sim n$)は、図7(a)に示すデータ構造を持ち、自サーバ内のファイル毎の負荷状況を示す情報と、そのファイルの属性を示す情報(ファイル属性)と、レプリケーションフラグを含むファイル別負荷状況情報を保持する。ファイル属性は、対応するファイルがオリジナルであるかレプリカ(複製)であることを示す。また、レプリケーションフラグは、対応するフラグがオリジナルの場合、そのレプリカを他サーバ側に生成済みであるか否か、つまりレプリケーション済みであるか否かを示す。

【0069】図6には、ファイルサーバ10-nが持つストレージ装置12-n内に、ファイルサーバ10-1が持つストレージ装置12-1内の任意のファイル611のレプリカ612がレプリケーションB1により保持されている様子が示されている。

【0070】次に、図6の構成の動作を説明する。仮想分散ファイルシステム61上の各仮想分散ファイルモジュール410-1~410-nは、ファイル別負荷状況情報保持部16-1~16-nを例えば定期的に参照する。そしてモジュール410-1~410-nは、保持部16-1~16-nに保持されているファイル別の負荷状況情報から、自サーバ(のストレージ装置12-1~12-n)に保持されているファイルの中に、第1の閾値を超えた負荷のファイルが存在することを検出した場合、他サーバの1つに対して対応するファイルのレプリカを非同期に生成するレプリケーション動作を、例えばプライベート通信路5を介してのサーバ間通信により行う。ここでファイルの負荷状況は、当該ファイルへの要求の待ち行列(キュー)にある要求数、或いは当該ファイルの待ち状態にある要求の示すサイズの総和であり、要求を受け付ける毎と要求を処理し終える毎に更新される。また、レプリケーションの対象サーバには、図示せぬサーバ情報保持部に保持されているサーバ情報に基づいて、例えば負荷が最も低いサーバを選択すればよい。

【0071】仮想分散ファイルモジュール410-1~410-nはレプリケーション動作を行うと、自サーバのファイル別負荷状況情報保持部16-1~16-nに保持されている対応するファイルの負荷状況情報中のレプリケ

ションフラグをレプリケーション済みの通知状態にセットする。またレプリケーション動作の対象となったサーバの仮想分散ファイルモジュールは、自サーバのファイル別負荷状況情報保持部内に対応するレプリカの負荷状況情報を追加する。

【0072】ここでは、図6に示すように、ファイルサーバ10-1の保持するファイル611のレプリケーションB1がプライベート通信路5を介してファイルサーバ10-nに対して行われて、そのレプリカ612が当該ファイルサーバ10-nのストレージ装置12-nに保持されたものとする。この場合、ファイルサーバ10-1のファイル別負荷状況情報保持部16-1に保持されているファイル611の負荷状況情報中のレプリケーションフラグがレプリケーション済みを示す状態にセットされる。また、ファイルサーバ10-nのファイル別負荷状況情報保持部16-nには、ファイル611のレプリカ612についての新たな負荷状況情報が追加される。この負荷状況情報中のファイル属性は、対応するファイルが(ファイル611の)レプリカ(612)であることを示す。

【0073】以後、クライアント2からファイル611の新たな読み出し要求があった場合、当該ファイル611を保持するファイルサーバ10-1(の仮想分散ファイルモジュール410-1)は、当該ファイル611の負荷が第2の閾値(但し、第2の閾値<第1の閾値)を超えているか否かを調べ、超えているならば、クライアント2からの要求に応答しない。この場合、クライアント2からの要求に対しては、レプリケーションを受けたファイルサーバ10-nが応答する。ここでファイルサーバ10-nは、ファイルサーバ10-1が応答するか否かを考慮する必要はなく、要求されたファイル611のレプリカ612を有する限り、クライアント2に応答すればよい。

【0074】このように、クライアント2からのファイル611に対する新たな読み出し要求を、そのレプリカ612を用いてファイルサーバ10-nが処理することで、そのファイル611を保持するファイルサーバ10-1では、それ以前に受け付けた当該ファイル611に対する読み出し要求の処理が進み、当該ファイル611の負荷が上記第2の閾値以下となる。するとファイルサーバ10-1上の仮想分散ファイルモジュール410-1は、ファイルサーバ10-n上の仮想分散ファイルモジュール410-nに対して、ファイル611のレプリカ612を消去するための要求を例えばプライベート通信路5を介したサーバ間通信により送る。

【0075】この要求を受けたファイルサーバ10-n上の仮想分散ファイルモジュール410-nは、既に受け付け済みの要求に対してのみレプリカ612を用いて処理を行い、しかる後にレプリカ612と対応する負荷状況情報を消去する。一方、ファイルサーバ10-1上の仮想分散ファイルモジュール410-1は、クライアント2か

らのファイル611に対する新たな読み出し要求があれば、それに対して応答する。

【0076】ところで、ファイルサーバ10-1からファイルサーバ10-nへのファイル611のレプリケーションにより、当該ファイル611に対する読み出し要求をファイルサーバ10-nで受け付けるようになった結果、ファイルサーバ10-1におけるファイル611の負荷が第2の閾値以下となる前に、ファイルサーバ10-nにおける当該ファイル611のレプリカ612の負荷が第1の閾値を超えることがあり得る。

【0077】そこで、このような場合、今度はファイルサーバ10-nがレプリカ612を用いて次の世代のレプリカを他の1つのサーバに生成し、即ちレプリケーションのレプリケーションを行い、そのサーバでファイル611に対する読み出し要求を処理させればよい。そのためには、ファイル別負荷状況情報保持部16-i (i=1~n) に保持されるファイル毎の負荷状況情報に、図7(a)に示したような負荷状況とレプリケーションフラグに加えて、図7(b)に示すように、ファイルの世代情報を持たせるとよい。

【0078】この場合、ある世代のレプリカの負荷が上記第2の閾値以下に下がった時点で、当該レプリカを持つサーバから、そのサーバによるレプリケーションの対象となったサーバの持つ次世代のレプリカを消去する等の制御を行うことができる。このとき、レプリカの消去が要求されたサーバが、別のサーバに対して更に次世代のレプリカを生成している場合、その更に次世代のレプリカを消去するとよい。この他に、少なくともレプリカに関連したファイルの負荷状況情報については、前記したサーバ情報と同様に、各サーバ間の一致化を図ることにより、同一ファイル（レプリカを含む）について、負荷が最も低いファイルを持つサーバが、当該ファイルに対する読み出し要求に応答するようにしてもよい。

【0079】最近では、ビデオ、オーディオ等のストリーミングデータや、或いはWWW (World Wide Web) のコンテンツ等、基本的には読み出しが主で、比較的サイズが大きく、ある程度のレスポンス（場合によっては帯域保証）が必要なデータが増加しつつある。しかもこうしたデータは、短期的に見て特定のデータ（ファイル）にアクセスが集中するケースが想定されるため、レスポンスを確保するのが困難な場合もある。以上に述べた図6の構成は、こうした状況を想定したもので、特定のファイルにアクセスが集中した場合に、自動的に当該ファイルのレプリケーションを行うことで、当該ファイルへのアクセスを分散させることができるようにしている。この構成は、単に負荷分散だけでなく、例えば重要性の高いファイルのバックアップに利用することも可能である。

【0080】〔第5の実施形態〕図8は本発明の第5の実施形態に係る仮想分散ファイルサーバシステムを適用

するコンピュータ・ネットワークシステムの構成を示すブロック図であり、図4と同一部分には同一符号を付してある。

【0081】図8において、8は図4中の仮想分散ファイルサーバシステム4に相当する仮想分散ファイルサーバシステムである。この仮想分散ファイルサーバシステム8の特徴は、ファイルサーバ10-1~10-n、及びストレージ装置12-1~12-nが、例えばFC-AL (Fibre Channel Arbitrated Loop) 80により相互接続され、（ホストとしての）各ファイルサーバ10-1~10-nから（ターゲットとしての）ストレージ装置12-1~12-nの共有が可能な（つまりマルチホスト可能な）ネットワーク構成を適用している点にある。ここでは、図4の構成と異なって、プライベート通信路5を持たない点に注意されたい。

【0082】この図8の構成では、図4の構成において（仮想分散ファイルモジュール410-1~410-nの通信モジュール413-1~413-nにより）プライベート通信路5を介して行われるサーバ間通信を、図1の構成と同様にネットワーク3を介して行えばよい（図は、この状態が示されている）。また、上記サーバ間通信を、ファイルサーバ10-1~10-nのストレージ接続用のインタフェースを介してFC-AL 80上で行うようにしてもよい。この場合、プライベート通信路5を用いたのと同様に、ネットワーク3の負荷を軽減できる。

【0083】図8の構成によれば、ストレージ装置12-1~12-nが全てのファイルサーバ10-1~10-nから直接に見えるので、各サーバ10-1~10-nに図6中のファイル別負荷状況情報保持部16-1~16-nを持たせることで、前記第4の実施形態で述べたようなレプリケーション動作や負荷分散を容易に行うことができる。なお、マルチホスト可能なネットワーク（インタフェース）はFC-AL 80に限るものではなく、SCSI (Small Computer System Interface) バスであっても構わない。

【0084】

【発明の効果】以上詳述したように本発明によれば、ネットワーク上に分散した複数のファイルサーバを、クライアントからは単一のサーバとして扱うことができ、サーバ台数やストレージ装置の接続状態をクライアントに意識させることがない。

【0085】また本発明によれば、サーバを増設した場合、動的にボリュームを拡張することもできる。

【0086】更に本発明によれば、複数のサーバ間で自律的な負荷分散が実現できる。

【図面の簡単な説明】

【図1】本発明の第1の実施形態に係る仮想分散ファイルサーバシステムを適用するコンピュータ・ネットワークシステムの構成を示すブロック図。

【図2】図1中のマッピングテーブルのデータ構造例を

示す図。

【図3】図1中のサーバ情報保持部のデータ構造例を示す図。

【図4】本発明の第2の実施形態に係る仮想分散ファイルサーバシステムを適用するコンピュータ・ネットワークシステムの構成を示すブロック図。

【図5】本発明の第3の実施形態に係る仮想分散ファイルサーバシステムを適用するコンピュータ・ネットワークシステムの構成を示すブロック図。

【図6】本発明の第4の実施形態に係る仮想分散ファイルサーバシステムを適用するコンピュータ・ネットワークシステムの構成を示すブロック図。

【図7】図6中のファイル別負荷状況情報保持部のデータ構造例を示す図。

【図8】同実施形態の動作を説明するタイミングチャート。

【符号の説明】

- 1, 4, 6, 8…仮想分散ファイルサーバシステム
2…クライアント

3…ネットワーク

5…プライベート通信路

10-1～10-n…ファイルサーバ

11, 41, 61…仮想分散ファイルシステム

12-1～12-n…ストレージ装置

13-1～13-n…ローカルファイルシステム

14-1～14-n…マッピングテーブル

15-1～15-n…サーバ情報保持部

16-1～6-n…ファイル別負荷状況情報保持部

80…F C - A L (マルチホスト可能なインタフェース)

110-1～110-n, 410-1～410-n…仮想分散ファイルモジュール (管理モジュール)

111-1～111-n…仮想分散ファイルインタフェース

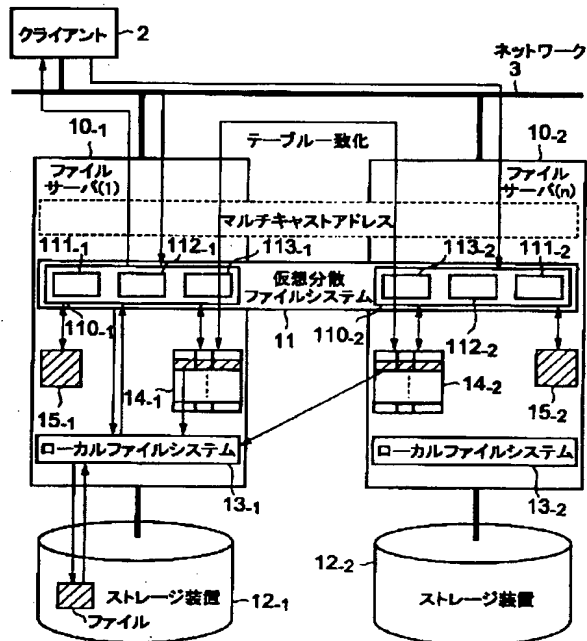
112-1～112-n…ローカルファイルインタフェース

113-1～113-n…通信モジュール

611…ファイル

612… (ファイル611の) レプリカ

【図1】



- 1
仮想分散ファイル
サーバシステム
14-1, 14-2…マッピングテーブル
15-1, 15-2…サーバ情報保持部
110-1, 110-2…仮想分散ファイルモジュール
111-1, 111-2…仮想分散ファイルインタフェース
112-1, 112-2…ローカルファイルインタフェース
113-1, 113-2…通信モジュール

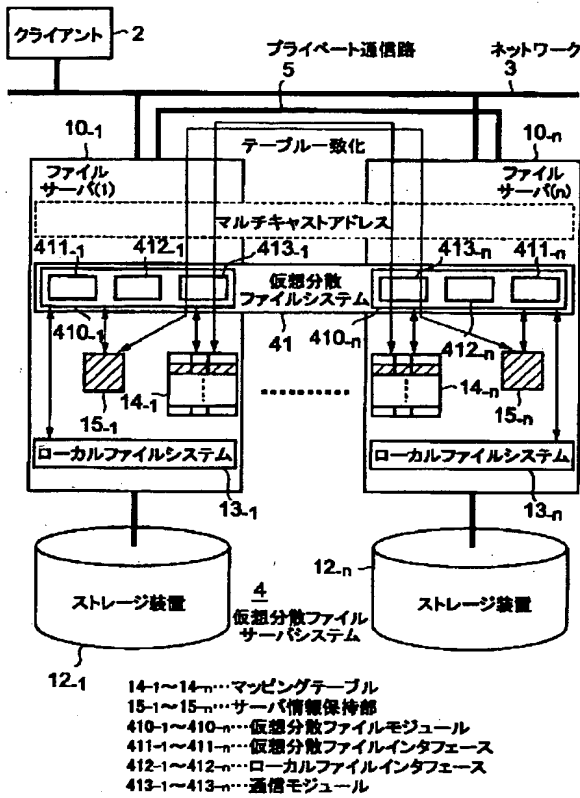
【図2】

141	142	143	144	145	145
ファイル名	仮想パス	所在位置情報	パーミッション情報	その他の属性	その他の属性
ファイル名	仮想パス	所在位置情報	パーミッション情報	その他の属性	その他の属性
ファイル名	仮想パス	所在位置情報	パーミッション情報	その他の属性	その他の属性

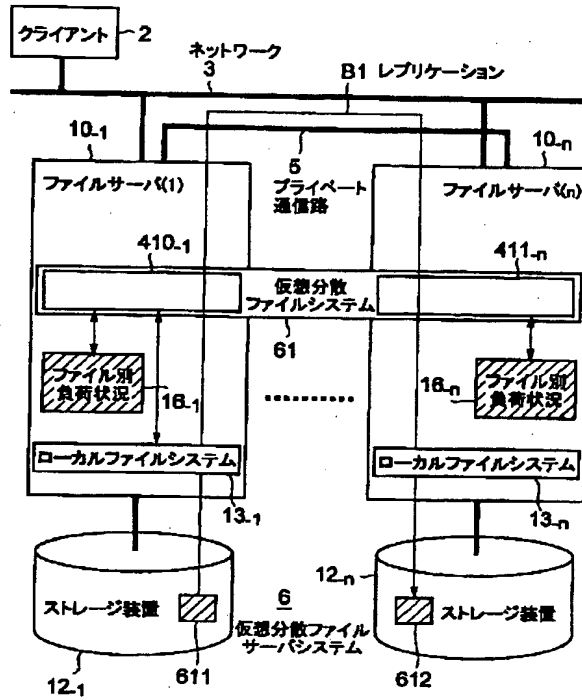
【図3】

サーバID	リソース情報	負荷状況
サーバ1		
サーバ2		

【図 4】

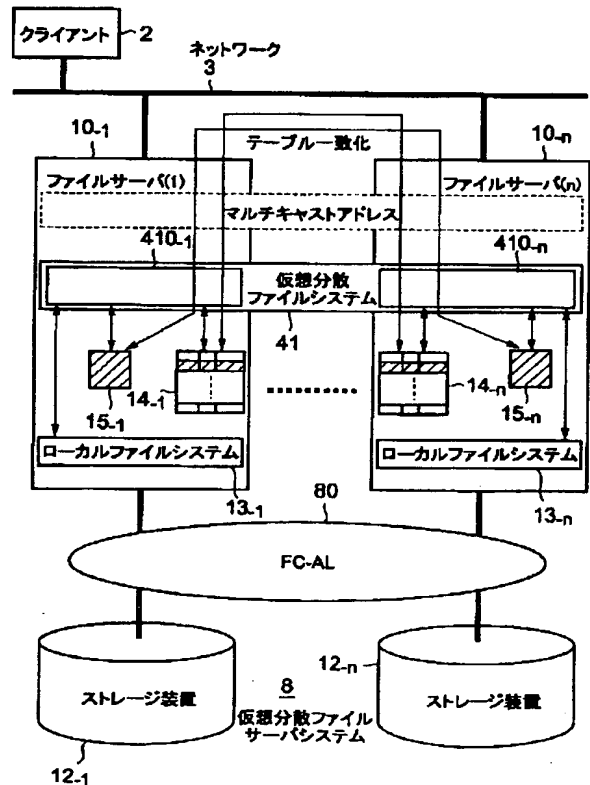
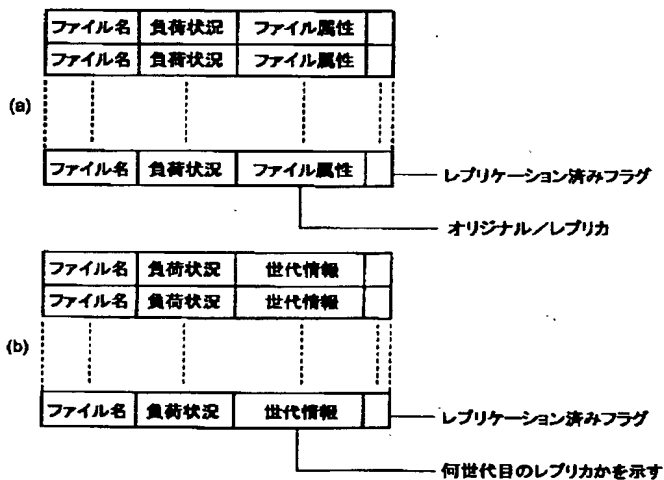


【図 6】

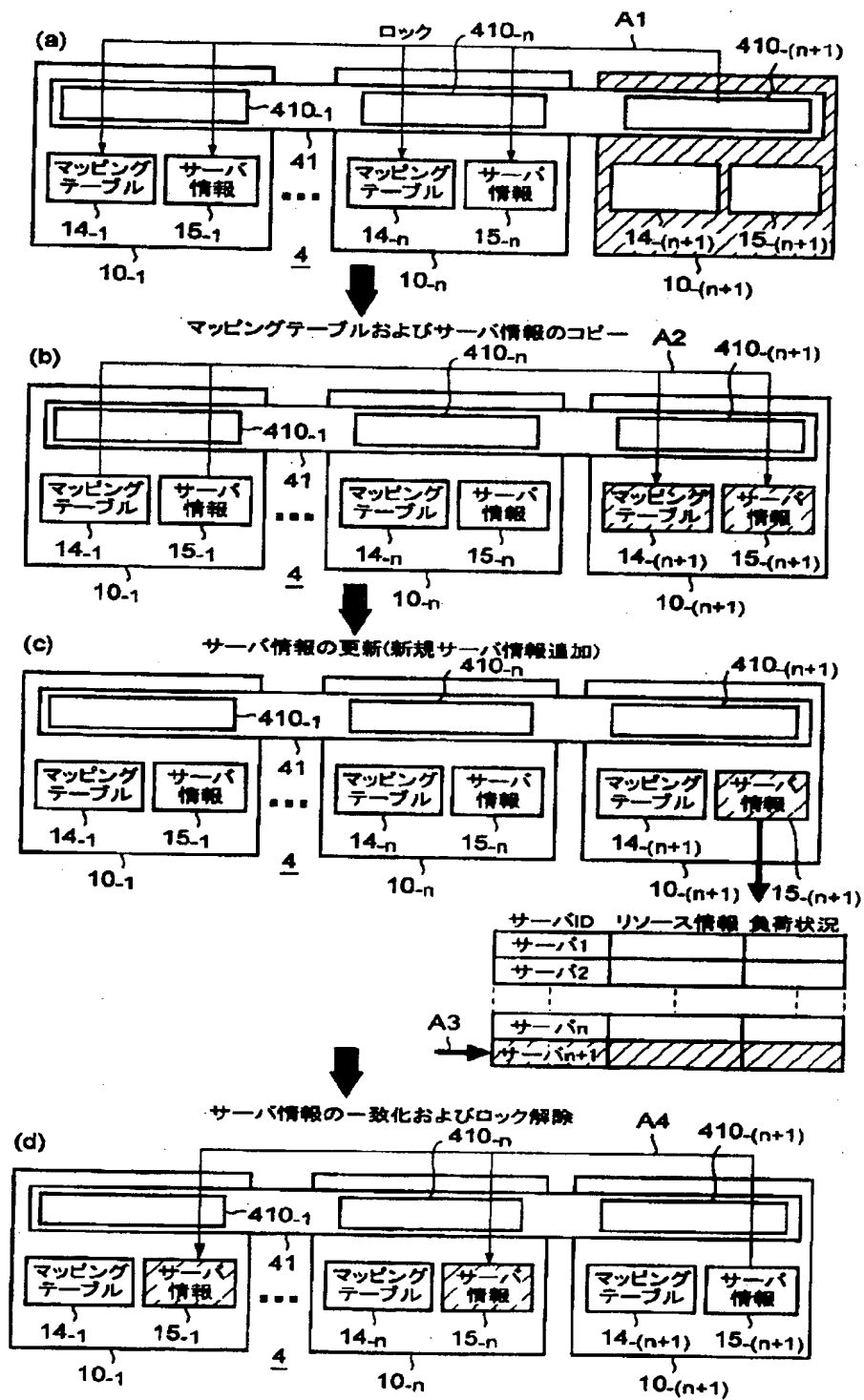


【図 8】

【図 7】



【図5】



フロントページの続き

Fターム(参考) 5B082 CA18 EA07 HA03 HA05 HA08
HA09
5B089 GA12 JA11 JB15 KA00 KC15
KC28 KE07

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-207370

(43)Date of publication of application : 28.07.2000

(51)Int.Cl.

G06F 15/177

G06F 12/00

G06F 15/16

(21)Application number : 11-011513

(71)Applicant : MATSUSHITA ELECTRIC IND CO LTD

(22)Date of filing : 20.01.1999

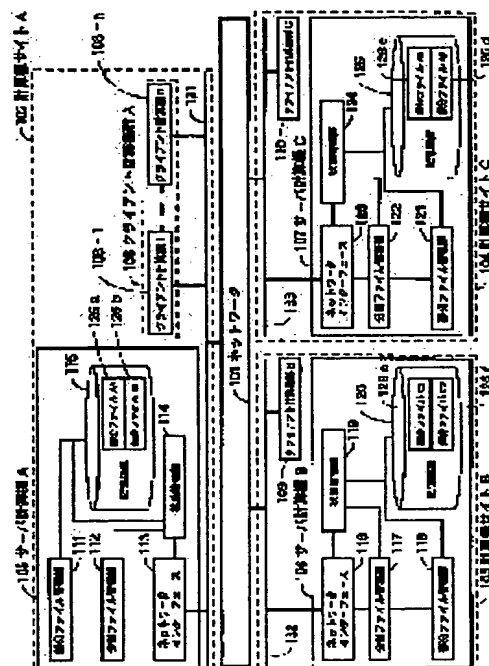
(72)Inventor : SATO MASAKI
UESUGI AKIO
YASUKOCHI RYUJI
TANAKA NORIKO

(54) DISTRIBUTED FILE MANAGEMENT DEVICE AND DISTRIBUTED FILE MANAGEMENT SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a distributed file management system which can make appropriate load distribution by means of plural server computers for generating, referring to and updating files.

SOLUTION: A distributed file management system is provided with server computers A, B, and C, client computer groups 108-110, and a network 101. The server computer A 105 is constituted of a storage device 115 which records partial files, a network interface 113, a partial file management section 111 which controls the write and read of the partial files, a status management section 114 which holds load information, and a distributed file management section 112. Since the arrangement of the partial files is determined, based on the load information of each server computer A, B, and C, the concentration of loads to a specific server computer can be avoided.



LEGAL STATUS

[Date of request for examination]

29.01.2002

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. *** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] Distributed file management equipment connected to the network where two or more server computers, 1, or two or more client computers which have a storage means characterized by providing the following to memorize data are connected. The state management tool which holds and manages the load information on two or more aforementioned server computers. A distributed file management means to determine the server computer which processes the aforementioned partial file based on the aforementioned load information which specifies the partial file of the aforementioned distributed file and is managed with the aforementioned state management tool corresponding to the processing demand of the distributed file from the aforementioned client computer.

[Claim 2] The aforementioned state management tool is distributed file management equipment according to claim 1 characterized by what it has for the external state management tool which holds the load information on the server computer which notified the aforementioned load information to other distributed file management equipments, and was notified from other distributed file management equipments as external load information.

[Claim 3] The aforementioned external state management tool is distributed file management equipment according to claim 2 characterized by what multicasting notifies the aforementioned load information for to distributed file management equipment besides the above.

[Claim 4] The aforementioned external state management tool is distributed file management equipment according to claim 2 or 3 characterized by what the aforementioned load information is notified for to other distributed file management equipments which adjoin among distributed file management equipment besides the above.

[Claim 5] Distributed file management system equipped with the network which connects two or more server computers characterized by providing the following, 1 or two or more client computers, two or more aforementioned server computers and the above 1, or two or more client computers. Each of two or more aforementioned server computers is a storage means to memorize the partial file which constitutes a part or all of a distributed file. A distributed file management means to determine the state management tool which holds and manages load information, and the server computer which processes the aforementioned partial file based on the aforementioned load information which specifies the partial file of the aforementioned distributed file and is managed with the aforementioned state management tool corresponding to the processing demand of the distributed file from the aforementioned client computer.

[Claim 6] The aforementioned state management tool notified the aforementioned load information to other server computers, and was notified from other server computers --- being concerned --- others --- the distributed file management system according to claim 5 characterized by what it has for the external state management tool which holds the load information on a server computer as external load information.

[Claim 7] The aforementioned external state management tool is distributed file management system according to claim 6 characterized by what multicasting notifies the aforementioned load information for to a server computer besides the above.

[Claim 8] It is the distributed file management system according to claim 6 or 7 characterized by

what the aforementioned external state management tool notifies the aforementioned load information for to other server computers which belong to a predetermined server computer group among the above 1 or two or more server computer groups by carrying out the group division of two or more aforementioned server computers at 1 or two or more server computer groups.

[Claim 9] It is the distributed file management system according to claim 6 or 7 characterized by what the aforementioned external state management tool notifies the aforementioned load information for to other server computers belonging to the server computer group which adjoins among the above 1 or two or more server computer groups by carrying out the group division of two or more aforementioned server computers at 1 or two or more server computer groups.

[Claim 10] The aforementioned distributed file-management means is distributed file management system the distributed file-management equipment according to claim 2 to 4 characterized by what other server computers of a partial file and a movement place which move based on the access information for every aforementioned partial file, the aforementioned load information, and the aforementioned external load information are determined, and it has a distributed file move means move this partial file to this server computer for, 6, or given in nine.

[Claim 11] The aforementioned distributed file move means is the distributed file-management equipment according to claim 10 or the distributed file management system carry out what the load of the aforementioned storage means included in the aforementioned load information detects that it is size rather than a predetermined value, other server computers of a partial file and a movement place which move based on the aforementioned external load information and the aforementioned access information determine, and this partial file moves for to these server computers of other as the feature.

[Claim 12] The aforementioned distributed file move means is the distributed file-management equipment according to claim 10 or the distributed file management system carry out what the remaining capacity of the aforementioned storage means included in the aforementioned load information detects that it is smallness from a predetermined value, and determines other server computers of a partial file and a movement place which move based on the aforementioned external load information and the aforementioned access information, and this partial file moves for to these server computers of other as the feature.

[Claim 13] The aforementioned distributed file move means is the distributed file-management equipment according to claim 10 or the distributed file management system which carries out as the feature in what the load of the aforementioned network included in the aforementioned load information detects that it is size rather than a predetermined value, other server computers of a partial file and a movement place which move based on the aforementioned external load information and the aforementioned access information determine, and this partial file moves to these server computers of other for.

[Claim 14] The aforementioned distributed file move means The aforementioned load information, the aforementioned external load information, the aforementioned access information, And it is based on the initial entry between the aforementioned client computer and two or more aforementioned server computers. It asks for the communication cost between the server computer which has a storage means to hold the aforementioned partial file, and the client computer which performed the processing demand. Distributed file management equipment according to claim 10 or distributed file management system characterized by what other server computers which serve as communication cost of smallness from this communication cost are determined, and the aforementioned partial file is moved for to these server computers of other.

[Claim 15] The aforementioned distributed file move means is the distributed file management equipment according to claim 10 to 14 or distributed file management system characterized by what it checks beforehand whether movement of the aforementioned partial file is possible for to the server computer besides the above used as the movement place of the aforementioned partial file.

[Claim 16] The aforementioned distributed file move means is the distributed file management equipment according to claim 10 to 15 or distributed file management system characterized by what other partial files are moved for to the aforementioned server computer from a server

computer besides the above when the aforementioned partial file is moved to a server computer besides the above.

[Claim 17] The aforementioned distributed file move means is the distributed file management equipment according to claim 10 to 16 or distributed file management system which makes a list the candidate of other server computers which can move the aforementioned partial file, and is characterized by what other server computers which move the aforementioned partial file are determined for based on this list.

[Claim 18] The aforementioned distributed file move means is the distributed file management equipment according to claim 10 to 17 or distributed file management system characterized by what the information about the server computer which created the aforementioned partial file with movement of the aforementioned partial file is sent for to a server computer besides the above.

[Claim 19] The aforementioned distributed file-management means is distributed file-management equipment or distributed file management system the claim 2 characterized by what other server computers of a partial file and a copy place to copy determine based on the access information for every aforementioned partial file, the aforementioned load information, and the aforementioned external load information, and it has a distributed file copy means copy the aforementioned partial file to a server computer besides the above for, 4 and 6, or given in 18.

[Claim 20] The aforementioned distributed file copy means is the distributed file-management equipment according to claim 19 or the distributed file management system characterized by what the load of the aforementioned storage means included in the aforementioned load information detects that it is size, determines other server computers of a partial file and a copy place which copy based on the aforementioned external load information and the aforementioned access information, and copies this partial file for to these server computers of other rather than a predetermined value.

[Claim 21] The aforementioned distributed file copy means is the distributed file-management equipment according to claim 19 or the distributed file management system characterized by what the load of the aforementioned network included in the aforementioned load information detects that it is size, determines other server computers of a partial file and a movement place copied based on the aforementioned external load information and the aforementioned access information, and copies this partial file for to these server computers of other rather than a predetermined value.

[Claim 22] The aforementioned distributed file copy means The aforementioned load information, the aforementioned external load information, It is based on the aforementioned access information and the initial entry between the aforementioned client computer and two or more aforementioned server computers. It asks for the communication cost between the server computer which has a storage means to hold the aforementioned partial file, and the client computer which performed the processing demand. Distributed file management equipment according to claim 19 or distributed file management system characterized by what other server computers which serve as communication cost of smallness from this communication cost are determined, and the aforementioned partial file is copied for to these server computers of other.

[Claim 23] The aforementioned distributed file copy means is the distributed file management equipment according to claim 19 to 22 or distributed file management system characterized by what it checks beforehand whether the copy of the aforementioned partial file is possible for to the server computer besides the above used as the copy place of the aforementioned partial file.

[Claim 24] The aforementioned distributed file copy means is the distributed file management equipment according to claim 19 to 23 or distributed file management system characterized by what other partial files are copied for to the aforementioned server computer from a server computer besides the above when the aforementioned partial file is copied to a server computer besides the above.

[Claim 25] The aforementioned distributed file copy means is the distributed file management equipment according to claim 19 to 24 or distributed file management system which makes a list the candidate of other server computers which can copy the aforementioned partial file, and is

characterized by what other server computers which copy the aforementioned partial file are determined for based on this list.

[Claim 26] The aforementioned distributed file copy means is the distributed file management equipment according to claim 19 to 25 or distributed file management system which are characterized by what other server computers of two or more copy places which copy the aforementioned partial file are chosen, and the aforementioned partial file is simultaneously copied for to two or more of other aforementioned selected server computers by multicasting.

[Claim 27] The aforementioned load information managed with the aforementioned state management tool is the distributed file management equipment according to claim 1 to 26 or distributed file management system characterized by what the capacity of the aforementioned storage means, a load, and the communication load between the aforementioned network and two or more aforementioned server computers are included for.

[Claim 28] Furthermore, the aforementioned server computer is the distributed file management equipment according to claim 1 to 27 or distributed file management system characterized by what it has for the partial file management means which writes the aforementioned partial file in the aforementioned storage means, and reads the aforementioned partial file from the aforementioned storage means.

[Claim 29] When the aforementioned processing demand from the aforementioned client computer is a creation demand of a distributed file, the aforementioned distributed file management means Divide this distributed file into two or more partial files, and the server computer holding the partial file which divided is determined based on the aforementioned load information managed with the aforementioned state management tool. When the aforementioned processing demand from the aforementioned client computer is a reference demand or updating demand of a distributed file The existence of a partial file set as the object of processing of the aforementioned reference demand or the aforementioned updating demand is decided.

Distributed file management equipment according to claim 1 to 28 or distributed file management system characterized by what the server computer which processes the aforementioned processing demand is determined for based on the aforementioned load information managed with the aforementioned state management tool.

[Claim 30] The aforementioned distributed file management means is the distributed file management equipment according to claim 1 to 29 or distributed file management system characterized by what it has a partial file size determination means to determine the size of the aforementioned partial file which constitutes a part or all of a distributed file for based on the information from the aforementioned client computer.

[Claim 31] The aforementioned distributed file management means is the distributed file management equipment according to claim 1 to 29 or distributed file management system characterized by what it has a partial file size determination means to determine the size of the aforementioned partial file which constitutes a part or all of the aforementioned distributed file for based on the kind of data currently recorded on the distributed file.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[The technical field to which invention belongs] This invention relates to the distributed file management equipment and distributed file management system which distribute and manage a file to two or more terminals in a computer network system. In the server client type computer network system which connected two or more server computers and client computers especially in the network, it is related with the distributed file management equipment and distributed file management system which distribute and manage a file by two or more server computers.

[0002]

[Description of the Prior Art] There is a distributed file system currently indicated by for example, JP 8-77054A public relations as distributed file management technology applied to the server client type computer network system (only henceforth a "network system") which connected two or more server computers and client computers in the network from the former. [0003] Drawing 30 shows the conventional distributed file system currently indicated by JP 8-77054A public relations. This conventional distributed file system is equipped with the network 3001 which connects two or more server computers 3002, 3003, 3004, and 3005, two or more client computers 3006, 3007, and 3008, and the server computers 3002, 3003, 3004, and 3005 and two or more client computers 3006, 3007, and 3008 of these plurality in drawing 30.

[0004] Here, the partial file B-1 (3002-3) of the partial file A-1 (3002-1) of the distributed file A and the distributed file B is held at the server computer 3002. Moreover, the partial file B-2 (3003-2) of the partial file A-2 (3003-1) of the distributed file A and the distributed file B is held at the server computer 3003. Moreover, the partial file B-3 (3004-2) of the partial file A-3 (3004-1) of the distributed file A and the distributed file B is held at the server computer 3004. [0005] Moreover, the server computer 3005 is equipped with each partial file A-1 to A-3 currently held at each server computers 3002, 3003, and 3004, and the distributed file management section 3005-2 which manages B-1 to B-3. Reference/updating demand distribution information 3005-1 for performing the distribution to a reference demand or updating demand (only henceforth "reference/updating demand") of the partial file from the client computers 3006, 3007, and 3008 is held.

[0006] The distributed file creation section 3006-1 in which the client computer 3006, on the other hand, creates the partial file of a distributed file according to a distributed file creation demand. The updating demand distribution section 3006-2 which determines the whereabouts of the partial file of this distributed file based on reference/updating demand distribution information 3005-1 on the server computer 3005 according to the updating demand to a distributed file. According to the reference demand to a distributed file, it has the reference demand distribution section 3006-3 which determines the whereabouts of the partial file of this distributed file based on reference/updating demand distribution information 3005-1 on the server computer 3005. In addition, other client computers 3007 and 3008 have same composition.

[0007] According to the conventional distributed file system mentioned above, when the user of the client computer 3006 demands creation of a distributed file, the distributed file creation

section 3006-1 of the client computer 3006 determines the server computer which creates the partial file of the distributed file concerned based on the distribution conditions decided beforehand, and creates the partial file of a distributed file to this server computer, for example. And reference/updating demand distribution information 3005-1 that it expresses with which server computer whether the partial file was created simultaneously with creation of this partial file is generated. It is transmitted to the server computer 3005 through a network 3001, and this reference/updating demand distribution information 3005-1 is held by the server computer 3005.

[0008] Moreover, when the user of the client computer 3006, for example, performs reference/updating demand which refers to or updates a distributed file, the client computer 3006 performs the open demand of the corresponding distributed file first to the distributed file management section 3005-2 of the server computer 3005. The distributed file management section 3005-2 of the server computer 3005 transmits reference/updating demand distribution information 3005-1 about the distributed file concerned to the client computer 3006 through a network 3001 corresponding to the open demand of the distributed file from the client computer 3006. The updating demand distribution section 3006-2 of the client computer 3006 or the reference demand distribution section 3006-3 transmits reference/updating demand to the server computer holding the partial file of a distributed file based on reference/updating demand distribution information 3005-1 received from the server computer 3005.

[0009] Thus, in the conventional distributed file system, when two or more processing demands concentrate to one distributed file by dividing a distributed file into two or more partial files, and distributing the processing (creation, reference, updating) to a distributed file to processing of a partial file unit, a load can be distributed, without centralizing a load on one server computer.

[0010]

[Problem(s) to be Solved by the Invention] However, in the conventional distributed file system as shown in drawing 30, since the server computer which creates this partial file is determined based on the fixed distribution rule decided beforehand in case the partial file which constitutes a distributed file is created, it is not what took into consideration the load information on a realistic server computer in distribution of the file to a server computer. For this reason, the creation demand of a partial file may occur further to the server computer which access and processing concentrate and is high [a load] in fact. Therefore, the load of only a specific server computer may become large and there was a problem that a suitable load distribution was not performed by two or more server computers.

[0011] Moreover, it sets to the conventional distributed file system. Once it distributes a partial file to a server computer fixed based on the above-mentioned distribution rule After creation of a partial file after the distribution (i.e., the server computer which was able to be distributed) In order not to perform movement or the copy of a partial file which were created, when access to a specific partial file concentrated, there was a problem that the load by access could not be distributed.

[0012] Therefore, the purpose of this invention is offering the distributed file management equipment and distributed file management system which can perform a suitable load distribution by two or more server computers in creation of a file, reference, and updating.

[0013]

[Means for Solving the Problem] In order to solve the above-mentioned technical problem, the distributed file management equipment of the 1st mode concerning this invention It is distributed file management equipment connected to the network where two or more server computers, 1, or two or more client computers which have a storage means to memorize data are connected. The state management tool which holds and manages the load information on two or more server computers. Corresponding to the processing demand of the distributed file from client computer, the partial file of a distributed file is specified and it is characterized by having a distributed file management means to determine the server computer which processes a partial file, based on the load information managed with the state management tool.

[0014] In the distributed file management equipment concerning an above-mentioned this invention A state management tool can also be equipped with the external state management

tool which holds the load information on the server computer which notified load information to other distributed file management equipments, and was notified from other distributed file management equipments as external load information.

[0015] You may make it an external state management tool notify load information to other distributed file management equipments by multicasting here. Moreover, an external state management tool can also notify load information to other distributed file management equipments which adjoin among other distributed file management equipments.

[0016] In order to solve the above-mentioned technical problem, moreover, the distributed file management system of the 1st mode concerning this invention In the distributed file management system equipped with the network which connects two or more server computers, one or two or more client computers, two or more server computers and one or two or more client computers Each of two or more server computers A storage means to memorize the partial file which constitutes a part or all of a distributed file, State management tool which holds and manages load information It corresponds to the processing demand of the distributed file from a client computer. The partial file of a distributed file is specified and it is characterized by having a distributed file management means to determine the server computer which processes a partial file, based on the load information managed with the state management tool.

[0017] the distributed file management system concerning an above-mentioned this invention -- setting -- the state management tool notified load information to other server computers, and was notified from other server computers -- being concerned -- others -- you may make it have the external state management tool which holds the load information on a server computer as external load information Here, an external state management tool can also notify load information to other server computers by multicasting.

[0018] Moreover, it sets to the distributed file management equipment and distributed file management system concerning an above-mentioned this invention. Two or more server computers The group division is carried out at one or two or more server computer groups. An external state management tool You may make it notify load information to other server computers which belong to a predetermined server computer group among one or two or more server computer groups. Or you may make it an external state management tool notify the aforementioned load information to other server computers belonging to the server computer group which adjoins among one or two or more server computer groups.

[0019] Furthermore, a distributed file management means can determine other server computers of a partial file and a movement place which move based on the access information, load information, and external load information for every partial file, and can also be equipped with a distributed file move means to move this partial file to this server computer.

[0020] A distributed file move means detects that it is size here rather than a value predetermined in the load of the storage means included in load information. Other server computers of a partial file and a movement place which move based on external load information and access information are determined. this partial file -- these server computers of other -- moving -- you may make -- Or a distributed file move means detects that it is smallness from a value predetermined in the remaining capacity of the storage means included in load information. Other server computers of a partial file and a movement place which move based on external load information and access information are determined, and you may make it move this partial file to these server computers of other. Moreover, a distributed file move means detects that it is size rather than a value predetermined in the load of the network, included in load information. other server computers of a partial file and a movement place which move based on external load information and access information -- determining -- this partial file -- these server computers of other -- it can also move -- A distributed file move means Or load information, external load information, access information. And it is based on the initial entry between a client computer and two or more server computers. It asks for the communication cost between the server computer which has a storage means to hold the partial file, and the client computer which performed the processing demand, other server computers which serve as communication cost of smallness from this communication cost are determined, and you may make it move a partial file to these server computers of other.

[0021] Furthermore, a distributed file move means can also check beforehand whether movement of the aforementioned partial file is possible to other server computers used as the movement place of a partial file. When a distributed file move means moves a partial file to other server computers, it can also move other partial files to a server computer from other server computers again. Furthermore, a distributed file move means makes a list the candidate of other server computers which can move a partial file, and you may make it determine other server computers which move a partial file based on this list. Moreover, a distributed file move means can send the information about the server computer which created the partial file with movement of a partial file to other server computers.

[0022] Moreover, an above-mentioned distributed file management means can determine other server computers of a partial file and a copy place to copy based on the access information, load information, and external load information for every partial file, and can also be equipped with a distributed file copy means to copy a partial file to other server computers.

[0023] At this time, a distributed file copy means detects that it is size rather than a value predetermined in the load of the storage means included in load information. Other server computers of a partial file and a copy place copied based on external load information and access information are determined. You may make it copy this partial file to these server computers of other. A distributed file copy means The load of the network included in load information detects that it is size, and determines other server computers of a partial file and a movement place copied based on external load information and access information, and you may make it copy this partial file to these server computers of other rather than a predetermined value. Or a distributed file copy means asks for the communication cost between the server computer which has a storage means hold the partial file, and the client computer which performed the processing demand based on load information, external load information, access information, and the initial entry between a client computer and two or more server computers, determines other server computers which serve as communication cost of smallness from this communication cost, and can copy a partial file to these server computers of other.

[0024] Here, as for a distributed file copy means, it is good to check beforehand whether the copy of a partial file is possible to other server computers used as the copy place of a partial file. Moreover, when a distributed file copy means copies a partial file to other server computers, it may copy other partial files to a server computer from other server computers. Moreover, a distributed file copy means makes a list the candidate of other server computers which can copy a partial file, and can determine other server computers which copy a partial file based on this list. Furthermore, a distributed file copy means can choose other server computers of two or more copy places which copy a partial file, and can also copy a partial file to two or more other selected server computers simultaneously by multicasting.

[0025] Moreover, the load information managed with the state management tool can contain the capacity of a storage means, a load, and the communication load between a network and two or more server computers.

[0026] Furthermore, it sets to the distributed file management equipment and distributed file management system concerning an above-mentioned this invention. A server computer can also be equipped with the partial file management means which writes a partial file in a storage means, and reads a partial file from a storage means.

[0027] Moreover, distributed file management means When the processing demand from a client computer is a creation demand of a distributed file Divide this distributed file into two or more partial files, and the server computer holding the partial file which divided is determined based on the load information managed with the state management tool. When the processing demand from client computer is a reference demand or updating demand of a distributed file the existence of a partial file set as the object of processing of a reference demand or an updating demand is decided, and the server computer which processes a processing demand is determined based on the load information managed with the state management tool -- you may make it like

[0028] Moreover, a distributed file-management means can have a partial file size determination means determine the size of the partial file which constitutes a part or all of a distributed file,

based on the information from a client computer. Or a partial file size determination means determine the size of the partial file which constitutes a part or all of a distributed file may make a distributed file-management means have based on the kind of data currently recorded on the distributed file.

[0029] In the distributed file management equipment and distributed file management system concerning an above-mentioned this invention, since the server computer by which a distributed file management means arranges a partial file based on the load information on a server computer is determined, concentration of the load to a specific server computer is avoidable.

[0030] Moreover, since the server computer by which a distributed file management means arranges a partial file based on the load information on other server computers is determined, it is avoidable that a load concentrates by the specific server computer.

[0031] Moreover, the imbalance of concentration of the load to the storage means of a specific server computer and the capacity of a storage means is avoidable by moving a partial file to other server computers. Moreover, concentration of the load to the storage of a specific server computer is avoidable by copying a partial file to other server computers.

[0032] Moreover, since the size of the partial file which constitutes a distributed file can be changed suitably, it is avoidable to divide into two or more partial files the data which have relation logically and in content, for example, the data for one picture etc.

[0033]

[Embodiments of the Invention] Hereafter, the gestalt of operation of the distributed file management equipment of this invention and distributed file management system is explained using drawing 29 from drawing 1.

[0034] (Gestalt 1 of operation) Drawing 1 is the block diagram showing an example of the gestalt of operation of the 1st of the distributed file management system in this invention. It set to drawing 1 and this distributed file management system is equipped with the networks 101, such as a Local Area Network which connects mutually two or more computers site A102 equipped with the client computer group which consists of two or more client computers, such as server computers, such as a personal computer and a workstation, and a personal computer, and a workstation, the computer site B103 and the computer site C104, the computer site A102 and the computer site B103, and the computer site C104, and a Wide Area Network.

[0035] Here, the computer site A102 is equipped with two or more server computers (only "the server computer A105" is shown in drawing 1), such as a personal computer and a workstation, and the client computer group A108 which consists of client computer 1-n (108-1 - 108-n), such as a personal computer and a workstation. This computer site A102 has connected two or more server computers (only "the server computer A105" is shown in drawing 1) and client computer groups A108 in the internal networks 131, such as Ethernet, for example, has become the Internet domain.

[0036] Moreover, like the computer site A102, the computer site B103 was equipped with the client computer group B109 which consists of two or more server computers (only "the server computer B106" is shown in drawing 1), and two or more client computers, and the computer site C104 is equipped with the client computer group C110 which consists of two or more server computers (only "the server computer C107" is shown in drawing 1), and two or more client computers. Furthermore, these computer sites B103 and the computer site C104 -- the computer site A102 -- the same -- two or more server computers (in drawing 1, only the "server computer B106" and the "server computer C107" are shown), and the client computer groups B109 and the client computer groups C110 -- each -- it has connected in the internal network 132 and the internal network 133, for example, has become the Internet domain.

[0037] The storage 115, such as a hard disk with which the server computer A105 records the partial file of a distributed file. The network interface 113 for connecting with the internal networks 131, such as Ethernet. The partial file management section 111 which controls the writing and read-out to the storage 115 which is recording the partial file. With the state Management Department 114 which supervises the load to storage 115, the remaining capacity of storage 115, and the load to a network interface 113, and holds the information about such loads and capacity it is constituted by the partial file management section 111, the state

Management Department 114, and the distributed file management section 112 connected to the network interface 113.

[0038] This distributed file management section 112 directs writing and read-out of a partial file in the partial file management section 111. Moreover, when creating a distributed file, based on the information acquired from the state Management Department 114, the distributed file management section 112 divides a distributed file into two or more partial files, and determines the server computer which arranges each partial file (record). Moreover, in referring to or updating the distributed file created before, it determines the server (recorded) computer by which the partial file of the corresponding distributed file exists.

[0039] The server computer B106 and the server computer C107 have the same composition as the server computer A105. That is, the server computer B108 is constituted by storage 120, a network interface 118, the partial file management section 116, the state Management Department 119, and the distributed file management section 117. Moreover, the server computer C107 is constituted by storage 125, a network interface 123, the partial file management section 121, the state Management Department 124, and the distributed file management section 122.

[0040] Drawing 2 is drawing showing an example of the composition of a distributed file. The distributed file 201 is constituted by two or more partial files 202-1 - 202-n in drawing 2.

[0041] In drawing 1, the state after each distributed files A, B, and C were created is shown. That is, the partial file A1 (126a) of the distributed file A and the partial file B1 (126b) of the distributed file B are recorded on the storage 115 of the server computer A105. Moreover, the partial file C1 (126c) of the distributed file C and the partial file C2 (126f) of the distributed file C are recorded on the storage 120 of the server computer B108. Moreover, the partial file A2 (126c) of the distributed file A and the partial file A3 (126d) of the distributed file A are recorded on the storage 125 of the server computer C107.

[0042] Next, operation of the distributed file management system constituted as mentioned above is explained. The creation demand of the distributed file A is published to the server computer A105 from the client computer 1 (108-1) of the client computer group A108 below, and distributed processing in case the partial files A1-A3 as shown in drawing 1 are created is made into an example, and is explained. Here, the storage 115, 120, and 125 shown in drawing 1 shall have two or more storage sections or storage regions (only henceforth the "storage section"), respectively. Two or more of these storage sections may be one record medium physically, and may be two or more record media.

[0043] In drawing 1, the creation demand of the distributed file A is first published by the server computer A105 from the client computer 1 (108-1). The creation demand of this distributed file A is received by the distributed file management section 112 of the server computer A105 through the network interface 113 of the internal network 131 and the server computer A105.

[0044] Drawing 3 is a flow chart which shows the algorithm of the distributed file management section at the time of receiving the creation demand of a distributed file of operation. Hereafter, detailed operation of the distributed file management section 112 is explained using drawing 1 and drawing 3.

[0045] The distributed file management section 112 which received the creation demand of the distributed file A acquires first the load information which the state Management Department 114 has managed (Step 301).

[0046] Drawing 4 is drawing showing the load information table 401 managed at the state Management Department 114. In drawing 4, the load information table 401 consists of the storage load information network load information table 402 and 403.

[0047] This storage load information table 402 consists of items of the "remaining capacity" which indicates the remaining capacity [Mbytes] of each storage section to be the "load" which indicates the load information on each storage section [X] to be the "storage identifier" for discriminating two or more storage sections of the storage 115 connected to the server computer A105. Here, the "load" of each storage section of storage 115 shows what % of the maximum transfer rates of each storage section of storage 115 is used.

[0048] Moreover, the network load information table 403 The data sent out on a network 101

through a network interface 113 [what bandwidth (use communication-band width of face [Mbps]) is used by being sent out towards which computer site (sending-out place site), and] Moreover, the received data are sent from which computer site (sending out agency site), and it is shown whether it has received using the bandwidth (use communication-band width of face [Mbps]) of how much. The item of this "sending out agency site" shows the computer site of the sending-out origin of data, and the item of a "sending-out place site" shows the computer site of the sending-out place of data. That is, in SiteA, SiteB shows the computer site B106 and SiteC shows the computer site C107 for the computer site A105. Here, a transmitting agency site and a transmission place site are named generically, and it is called a network link (only henceforth a "link"). Moreover, the item of "use communication-band width of face" shows the communication-band width of face [Mbps] currently used between the sending out agency site and the sending-out place site.

[0049] That is, when the distributed file management section 112 receives the creation demand of the distributed file A, the distributed file management section 112 acquires the information that the load of the storage section of the storage identifier DiskID1 is 20 [%], and remaining capacity is 10 [Mbytes] based on the information on the storage load information table 402 gained from the state Management Department 114, as shown in drawing 4 (Step 301).

[0050] Next, based on the information on the storage load information table 402 obtained from the state Management Department 114, out of the storage 115 connected to the server computer A105, remaining capacity fully remains, and a load chooses the low storage section from a predetermined threshold, and the distributed file management section 112 assigns the partial file in order to this storage section. Here, as a threshold, it is good to use 80 [%] etc. for example. However, this threshold can be suitably determined according to the composition of storage 115 etc. Moreover, the size of a partial file has a desirable fixed length, and it is good to make it the same size by all server computers. At this time, when all the partial files A1-A3 are assigned by the storage section of storage 115, Step 304 is processed. On the other hand, Step 303 is processed when all the partial files A1-A3 are not assigned by the storage section of storage 115 (Step 302).

[0051] In the case of the gestalt (drawing 1) of this operation, all the partial files A1-A3 are assigned to the storage section of storage 115 -- having not had (Step 302) -- the partial file A1 (126a) of the distributed file A is assigned to the server computer A105, and the remaining partial files A2 and A3 are assigned to other server computers C107

[0052] The distributed file management section 112 asks whether the partial files A2 and A3 can be created through the internal network 131 and a network 101 to other server computers, in order to assign the partial files A2 and A3 which were not assigned at Step 302 to other server computers. By other server computers which received the inquiry, the load information table 401 of the state Management Department of self is investigated, and it answers [whether creation of the partial files A2 and A3 is possible, and] (Step 303). The distributed file management section of each server computer will perform an exchange of an inquiry of creation of this partial file, and the signal of an answer through a network interface and a network.

[0053] In the case of the gestalt (drawing 1) of this operation, the distributed file management section 112 asks the distributed file management section 122 of the server computer C107 of the computer site C104 whether creation of the partial file A2 (126c) and the partial file A3 (126d) can be performed through a network interface 113 and a network 101. Based on the load information table 401 obtained from the state Management Department 124, the distributed file management section 122 of the server computer C107 performs the same judgment as the above-mentioned step 302, and answers [whether creation of the partial file A2 (126c) and the partial file A3 (126d) can be performed, and] the server computer A105 (Step 303). In the case of drawing 1, to the storage 125 of the server computer C107, the partial file A2 (126c) and the partial file A3 (126d) can be created.

[0054] Next, the distributed file management section 112 of the server computer A registers the information for managing the partial file A1 assigned to the storage 115 of the server computer A, and the partial files A2 and A3 assigned to the storage 125 of other server computers C (Step 304).

[0055] Drawing 5 is drawing showing the distributed file control table 501. Moreover, drawing 6 is drawing showing the partial file control table 601. In drawing 5, the distributed file control table 501 consists of items of "the partial file identification child list of [for discriminating the partial file which constitutes the "distributed file identification child" and distributed file for discriminating a distributed file]." Moreover, in drawing 6, the partial file control table 601 consists of items of the "address" which shows the "partial file identification child" for discriminating a partial file, and the address of a partial file. Here, the "partial file identification child" who showed drawing 6 is equivalent to the "partial file identification child" who constitutes the "partial file identification child list" shown by drawing 5.

[0056] If it sees about the distributed file A, in drawing 5, it expresses that the distributed file A consists of a partial file A1 (126a), a partial file A2 (126c), and a partial file A3 (126d) with the case of the gestalt (drawing 1) of this operation, for example. Moreover, in drawing 6, the address of the partial file A1 (126a) is "file://siteA/serverA/DiskID1/(storage identifier DiskID1 of the server computer A105 of the computer site A102)." The address of the partial file A2 (126c) is "file://siteC/serverC/DiskID2/(storage identifier DiskID2 of the server computer C107 of the computer site C104)." The address of the partial file A3 (126d) expresses that it is "file://siteC/serverC/DiskID2/(storage DiskID2 of the server C107 of the computer site C104)."

[0057] Next, when creating the partial file A1 to the storage 115 of the server computer A, through the partial file management section 111, the distributed file management section 112 writes the data from the client computer 1 (108-1) in storage 115, and creates the partial file A1 of the distributed file A. When recording the partial files A2 and A3 on other server computers C107, moreover, the distributed file management section 112 Request record of the partial files A2 and A3 from the distributed file management section 122 of the server computer C107 which records, and, simultaneously with it, it directs to the client computer 1 (108-1) which performed the creation demand of the distributed file A. Immediate data are transmitted to the server computer C107 which performs record from the client computer 1 (108-1). By the server computer C107 which received the request, the distributed file management section 122 registers the partial files A2 and A3 to the partial file control table 601, thus, other server computers C107 top -- the partial file A -- 2 and 3 are created (Step 305)

[0058] above -- carrying out -- creation of the partial file A1 (126a) of the distributed file A -- the data from the client computer 1 (108-1) -- the storage 115 of the server computer A105 -- ***** -- it is carried out by things Moreover, creation of the partial file A2 (126c) of the distributed file A and the partial file A3 (126d) sends the data from the client computer 1 (108-1) to the *** server computer C107, and is performed by writing in the storage 125 of the server computer C107.

[0059] Moreover, in the above-mentioned step 303, by all other server computers, when creation of a partial file is impossible, the distributed file management section 112 notifies that creation of a distributed file went wrong to the client computer 1 (108-1) which performed the creation demand of a distributed file (Step 306).

[0060] On the other hand, at Step 302, when creation of all the partial files of a distributed file is made to the storage of the server computer of self, the distributed file management section 112 is registered into the partial file control table 601 having shown the information for managing the distributed file assigned to storage 115 in the distributed file control table 501 shown in drawing 5, and drawing 6 (Step 307).

[0061] Next, through the partial file management section 111, the distributed file management section 112 writes the data from the client computer 1 (108-1) in storage 115, and creates all partial files (Step 308).

[0062] As mentioned above, according to the distributed file management system of this invention, since creation of a distributed file is performed in consideration of the load of each server computer, a load distribution comes to be made appropriately.

[0063] Next, the case where distributed file reference / updating demand is published to the server computer A105 from the client computer 1 (108-1) in the client computer group A108 is explained.

[0064] First, reference/updating demand to the distributed file A published from the client computer 1 (108-1) is received by the distributed file management section 112 through a network interface 113 in the server computer A105.

[0065] Drawing 7 is a flow chart which shows the algorithm of the distributed file management section at the time of receiving reference/updating demand of a distributed file of operation. Hereafter, detailed operation of the distributed file management section 112 is explained using drawing 7. Moreover, in the following, concrete operation which made the example processing of reference/updating demand to the distributed file A is also explained.

[0066] First, the distributed file management section 112 is based on reference/updating demand of the distributed file from the client computer 1 (108-1), from the distributed file control table 501, and the partial file control table 601, specifies the partial file referred to / updated, and asks for the address of the partial file (Step 701).

[0067] Here, as for the distributed file management section 112, in reference/updating demand to the distributed file A from the client computer 1 (108-1), based on the distributed file control table 501, it turns out that the distributed file A is constituted by the partial file A1 (126a), the partial file A2 (126c), and the partial file A3 (126d). It is based on the partial file control table 601, moreover, the address of the partial file A1 (126a) is "file://siteA/ServerA/DiskID1/(storage identifier DiskID1 of the server computer A105 of the computer site A102)." The address of the partial file A2 (126c) is "file://siteC/ServerC/DiskID2/(storage identifier DiskID2 of the server computer C107 of the computer site C104)." It turns out that the address of the partial file A3 (126d) is "file://siteC/ServerC/DiskID2/(storage identifier DiskID2 of the server C107 of the computer site C104)."

[0068] The distributed file management section 112 judges whether all the partial files that perform reference/updating exist in the storage 115 of the server computer A105 of self, a part, or all partial files exist in other server computers from the address of the partial file obtained at Step 701 (Step 702).

[0069] Here, in the case of the distributed file A, it exists in storage 115 and, as for the partial file A1 (126a), it turns out that partial FAIRU A2 (126c) and the partial file A3 (126d) exist in the storage 125 of the server computer C107 of the computer site C104.

[0070] Next, when all partial files do not exist in the storage 115 of the server computer A105 of self (Step 702), based on the address of the partial file obtained at Step 701, existence of a partial file is confirmed to other server computers by which the partial file which performs reference/updating is recorded (Step 703).

[0071] Here, in the case of the distributed file A, existence of the partial file A2 (126c) and the partial file A3 (126d) is checked at the distributed file management section 122 of the server computer C107 of the computer site C104.

[0072] If existence of a partial file is checked at Step 703, the distributed file management section 112 will perform read-out (reference) of the partial file which exists in storage 115 through the partial file management section 111, and the writing (updating) to a partial file based on reference/updating demand from the client computer 1 (108-1), when the partial file which performs reference/updating exists in the storage 115 of the server computer A105 of self. Moreover, when the partial file which performs reference/updating exists in the storage of other server computers, the distributed file management section 112 requires the reference/renewal of an applicable partial file of the server computer holding the partial file which performs reference/updating. It can come, simultaneously the distributed file management section 112 directs to give reference/updating demand to the server computer holding the partial file to which the ***** client computer 1 (108-1) gives reference/updating demand for reference/updating directly (Step 704).

[0073] Here, in the case of the distributed file A, the partial file A1 (126a) exists in the server computer A105 of the computer site A102, and the partial file A2 (126c) and the partial file A3 (126d) exist in the server computer C107 of the computer site C104. As for reference/updating demand to the partial file A1 (126a), the distributed file management section 112 performs reference/update process to storage 115 through the partial file management section 111. On the other hand, reference/updating demand to the partial file A2 (126c) and the partial file A3

(126d) will be directly performed between the client computers 1 (108-1) and the server computers C107 which performed reference/updating demand.

[0074] Moreover, when existence of a partial file is not checked at Step 703, the distributed file management section 112 notifies what the reference/renewal of a distributed file failed in the client computer 1 (108-1) which performed reference/change request of a distributed file (Step 705).

[0075] On the other hand, at Step 702, when all partial files exist in the storage 115 of the server computer A105 of self, the distributed file management section 112 performs read-out (reference) of the partial file which exists in storage 115 through the partial file management section 111, and the writing (updating) to a partial file based on reference/updating demand from the client computer 1 (108-1) (Step 706).

[0076] As mentioned above, according to the gist of operation mentioned above, when the demand to a server computer from a client computer is creation of a distributed file, a distributed file is divided into two or more partial files, the server computer which creates each partial file based on the load information on a server computer is determined for every partial file, and creation processing of a distributed file is carried out. Moreover, when the demands from a client computer are the reference/renewal of a distributed file, the server computer by which the partial file which constitutes a distributed file exists is specified, and the partial file distributed and arranged on 1 or two or more server computers is treated as one distributed file from a client computer. Thus, concentration of the load to a specific server computer can be lost in the case of creation / reference / change request of the distributed file from a client computer to a server computer.

[0077] (Gestalt 2 of operation) Drawing 8 is the block diagram showing an example of the gestalt of operation of the 2nd of the distributed file management system in this invention. In this drawing 8, the same sign is given to the same composition as drawing 1. The distributed file management system shown in drawing 8 is equipped with the networks 101, such as a Local Area Network which connects mutually two or more computers site A802 equipped with the client computer group which consists of two or more client computers, such as server computers, such as a personal computer and a workstation, and a personal computer, and a workstation, the computer site B803 and the computer site C804, the computer site A802 and the computer site B803, and the computer site C804, and a Wide Area Network.

[0078] Here, the computer site A802 is equipped with two or more server computers (only "the server computer A805" is shown in drawing 8), such as a personal computer and a workstation, and the client computer group A108 which consists of client computer 1-n (108-1 ~ 108-n), such as a personal computer and a workstation. This computer site A802 has connected two or more server computers (only "the server computer A805" is shown in drawing 8) and client computer groups A108 in the internal networks 131, such as Ethernet, for example, has become the Internet domain.

[0079] Moreover, like the computer site A802, the computer site B803 was equipped with the client computer group B109 which consists of two or more server computers (only "the server computer B806" is shown in drawing 8), and two or more client computers, and the computer site C804 is equipped with the client computer group C110 which consists of two or more server computers (only "the server computer C807" is shown in drawing 8), and two or more client computers. Furthermore, these computer sites B803 and the computer site C804 --- the computer site A802 --- the same --- two or more server computers (in drawing 8, only the "server computer B806" and the "server computer C807" are shown), and the client computer groups B109 and the client computer groups C110 --- each --- it has connected in the internal network 132 and the internal network 133, for example, has become the Internet domain [0080] The storage 115, such as a hard disk with which the server computer A805 records the partial file of a distributed file. The network interface 113 for connecting with the internal networks 131, such as Ethernet. The partial file management section 111 which controls the writing and read-out to the storage 115 which is recording the partial file. With the state Management Department 814 which supervises the load to storage 115, the remaining capacity of storage 115, and the load to a network interface 113, and holds the information about such

loads and capacity it is constituted by the partial file management section 111, the state Management Department 814, and the distributed file management section 112 connected to the network interface 113.

[0081] This state Management Department 814 has the external state Management Department 811 holding the external load information which notified load information to other server computers, and was notified from other server computers.

[0082] The server computer B806 and the server computer C807 have the same composition as the server computer A805. That is, the server computer B806 is constituted by storage 120, a network interface 118, the partial file management section 116, the state Management Department 819 having the external state Management Department 812, and the distributed file management section 117. Moreover, the server computer C807 is constituted by storage 125, a network interface 123, the partial file management section 121, the state Management Department 824 having the external state Management Department 813, and the distributed file management section 122.

[0083] The difference with the distributed file management system shown in the distributed file management system shown in drawing 8 and drawing 1 here is a point equipped with the external state Management Department 811, 812, and 813 where the state Management Department 814, 819, and 824 which showed drawing 8 holds the external load information which notified load information to other server computers, and was notified from other server computers.

[0084] Drawing 9 shows an example of the external load information table 901 managed at the external state Management Department 811, 812, and 813. In drawing 9, the external load information table 901 consists of items of the "storage load information" which shows the load information on the storage of the server computer shown in the "server computer address" which shows the address of a server computer, and a server computer address. Moreover, "storage load information" consists of items of the "remaining capacity" which indicates the remaining capacity of storage to be the "load" which shows the "storage identifier" for discriminating storage, and the load of storage.

[0085] The external state Management Department 811, 812, and 813 performs the following operation, in order to notify the external load information from the external load information table 901 to other server computers.

[0086] First, the external state Management Department 811, 812, and 813 acquires the information on the storage load information table 402 managed at the state Management Department 814, 819, and 824 by having the change of state of the information which asks the information shown in the storage load information table 402 to the state Management Department 814, 819, and 824 to periodical or predetermined timing, or is shown in the storage load information table 402 from the state Management Department 814, 819, and 824 notified.

[0087] Next, the external state Management Department 811, 812, and 813 notifies storage load information to each server computer through each network interface 113, 118, and 123. By the server computer which received the notice, through network interfaces 113, 118, and 123, the external state Management Department 811, 812, and 813 receives storage load information, and records this information on each external load information table 901, respectively.

[0088] The case where the creation demand of the distributed file A is published to the server computer A805 about the distributed file management system constituted as mentioned above from the client computer 1 (108-1) in the client computer group A108 is explained as an example.

[0089] First, the creation demand of the distributed file A published from the client computer 1 (108-1) is received by the distributed file management section 112 through the network interface 113 of the internal network 131 and the server computer A805.

[0090] Drawing 10 is a flow chart which shows the algorithm of the distributed file management section when receiving the creation demand of a distributed file of operation. Hereafter, detailed operation of the distributed file management section is explained using drawing 10. In the gistalt of this operation, processing of Step 302 and Step 303 which are shown in drawing 3 explained with the gistalt of the 1st operation can be unified, and it can process at one step 1002.

[0091] First, the distributed file management section 112 acquires each information from the

load information table 401 which the state Management Department 814 has managed, and the external load information table 901 which the external state Management Department 811 has managed (Step 1001).

[0092] At this state Management Department 114, the load information table 401 as shown in drawing 4 is managed. The load information table 401 is constituted by the storage load information table 402 and the network load information table 403 in drawing 4.

[0093] This storage load information table 402 consists of items of the "remaining capacity" which indicates the remaining capacity [Mbytes] of each storage section to be the "load" which indicates the load information on each storage section [x] to be the "storage identifier" for discriminating two or more storage sections of the storage 115 connected to the server computer A105. Here, the "load" of each storage section of storage 115 shows what % of the maximum transfer rates of each storage section of storage 115 is used.

[0094] Moreover, the network load information table 403 The data sent out on a network 101 through a network interface 113 [what bandwidth (use communication-band width of face [Mbps]) is used by being sent out towards which computer site (sending-out place site), and] Moreover, the received data are sent from which computer site (sending out agency site), and it is shown whether it has received using the bandwidth (use communication-band width of face [Mbps]) of how much. The item of this "sending out agency site" shows the computer site of the sending-out place of data, and the item of a "sending-out place site" shows the computer site of the sending-out place of data. That is, in SiteA, SiteB shows the computer site B106 and SiteC shows the computer site C107 for the computer site A105. Here, a transmitting agency site and a transmission place site are named generically, and it is called a network link (only henceforth a "link"). Moreover, the item of "use communication-band width of face" shows the communication-band width of face [Mbps] currently used between the sending out agency site and the sending-out place site.

[0095] Moreover, the external state Management Department 811 has managed the external load information table 901 as shown in drawing 9.

[0096] Here, when creating the distributed file A, the "load" of the storage section of the storage 115 in which a "storage identifier" is shown by DiskID1 is 20 [%], and the distributed file management section 112 can acquire the information that "remaining capacity" is 10 [Mbytes] from the load information table 401 of the state Management Department 814. The distributed file management section 112 moreover, from the external load information table 901 of the external state Management Department 814 The "load" of the storage section of the storage 120 in which the "storage identifier" of the server computer B806 of the computer site B803 is shown by DiskID1 by 49 [%] The "load" of the storage section of the storage 125 in which "remaining capacity" is 1000 [Mbytes] and the "storage identifier" of the server computer C807 of the computer site C804 is shown by DiskID1 by 30 [%] The information that "remaining capacity" is 3000 [Mbytes] can be acquired.

[0097] Next, the distributed file management section 112 is first based on the storage load information table 402 obtained from the state Management Department 814. "Remaining capacity" out of each storage section of the storage 115 connected to the server computer A805 above a predetermined capacity And a "load" chooses the storage section which fulfills the conditions of being lower than a predetermined threshold, and assigns the partial file which divided the distributed file in order to each storage section which fulfills the conditions concerned. When all partial files are not assigned by storage 115, based on the information on the external load information table 901 that the distributed file management section 112 is obtained from the external state Management Department 811, "remaining capacity" of the storage section is more than a predetermined capacity, and a "load" chooses other server computers with the storage with which the storage section lower than a predetermined threshold exists, and the partial file which is not assigned yet -- being concerned -- others -- the storage section of the storage of a server computer -- order -- assigning -- this allocation -- being concerned -- others -- it notifies to a server computer And it judges whether all partial files assigned the storage section of each storage of a server computer, and have created (Step 1002).

[0098] Here, when creating the distributed file A, the distributed file management section 112 assigns the partial file A1 (126a) to the predetermined storage section of the storage 115 of the server computer A805 based on the storage load information table 401 of the state Management Department 814. Moreover, the distributed file management section 112 assigns the partial file A2 (126c) and the partial file A3 (126d) to each storage section of the storage 125 of the server computer C807 based on the storage load information on the external load information table 901 of the external state Management Department 811.

[0099] Thus, the server computer A805 can determine the server computer which arranges a partial file (storage), taking into consideration other loads and remaining capacity of a server computer without asking other server computers whether creation of a partial file is possible by acquiring the load of other server computers, and the information on remaining capacity from the external state Management Department 811.

[0100] Next, the distributed file management section 112 registers the management information of a distributed file into the distributed file control table 501 as shown in drawing 5, and the partial file control table 601 as shown in drawing 6 (Step 1003). In drawing 5, the distributed file control table 501 consists of items of "the partial file identification child list of [for discriminating the partial file which constitutes the "distributed file identification child" and distributed file for discriminating a distributed file]." Moreover, in drawing 6, the partial file control table 601 consists of items of the "address" which shows the "partial file identification child" for discriminating a partial file, and the address of a partial file. Here, the "partial file identification child" who showed drawing 6 is equivalent to the "partial file identification child" who constitutes the "partial file identification child list" shown by drawing 5.

[0101] Here, in the case of the distributed file A, in drawing 5, it means that the distributed file A consists of a partial file A1 (126a), a partial file A2 (126c), and a partial file A3 (126d).

Moreover, in drawing 6, the address of the partial file A1 (126a) is "file:///siteA/ServerA/DiskID1/(storage identifier DiskID1) of the server computer A105 of the computer site A102". The address of the partial file A2 (126c) is "file:///siteC/ServerC/DiskID2/(storage identifier DiskID2) of the server computer C107 of the computer site C104". The address of the partial file A3 (126d) expresses that it is "file:///siteC/ServerC/DiskID2/(storage identifier DiskID2) of the server computer C107 of the computer site C104".

[0102] Next, the distributed file management section 112 writes the data from the client computer 1 (108-1) in the predetermined storage section of storage 115 through the partial file management section 111, when recording a partial file on the storage section of storage 115. Moreover, in recording a partial file on the storage of other server computers, the distributed file management section 112 requests record of a partial file from the distributed file management section of the server computer which records. It can come, simultaneously it points to the distributed file management section 112 to the client computer 1 (108-1) which performed the creation demand of a distributed file, and directs to transmit immediate data to the server computer which records. By the server computer which received the request from the distributed file management section 112, the data of a partial file are received from the client computer 1 (108-1), and it records on the predetermined storage section of storage. Moreover, the distributed file management section of this server computer registers the information on a partial file to the partial file control table 601 of the partial file management section. Thus, a partial file is created on other server computers (Step 1004).

[0103] In the case of the distributed file A, creation of the partial file A1 (126a) is performed by writing the data from the client computer 1 (108-1) in the predetermined storage section of storage 115. Creation of the partial file A2 (126c) and the partial file A3 (126d) sends predetermined data to the direct server computer C807 from the client computer 1 (108-1), and is performed by writing in the predetermined storage section of the storage 125 of the server computer C807, respectively.

[0104] On the other hand, at Step 1002, when a partial file cannot be created to all the server computer, either, the distributed file management section 112 notifies that creation of a distributed file went wrong to the client computer 1 (108-1) which required distributed file creation (Step 1005).

[0105] As mentioned above, although creation of a distributed file was explained, about the case where reference/updating demand of a distributed file is published from a client computer to a server computer, it is the same as that of the case (drawing 7) of the gestalt of the 1st operation.

[0106] As mentioned above, it sets in the gestalt of operation of the 2nd of this invention. By having the external state Management Department 811, 812, and 813 where the state Management Department 814, 819, and 824 holds the external load information which notified load information to other server computers, and was notified from other server computers. The distributed file management sections 112, 117, and 122 can determine the server computer which arranges the partial file of a distributed file based on the load information on other server computers, and can avoid concentration of the load to a specific server computer.

[0107] In addition, in case storage load information is notified to each server computer from the external state Management Department 811, 812, and 813, it is good to use a unicast and multicasting. When multicasting is used especially, storage load information can be notified to all server computers all at once, and the traffic for a notice can be reduced.

[0108] Moreover, the server computer to notify is beforehand divided into two or more groups, and it can also notify to each of the server computer belonging to each group by the unicast, and can also notify by multicasting to each group. Thus, the traffic for a notice can be reduced. [0109] Furthermore, the server computer to notify can be limited to an adjoining server computer, i.e., the server computer connected directly in the network, and a unicast or multicasting can also notify. Thereby, the traffic for a notice can be reduced.

[0110] (Gestalt 3 of operation) Drawing 11 is the block diagram showing an example of the gestalt of operation of the 3rd of the distributed file management system in this invention. In drawing 11, the same sign is given to the same composition as drawing 8. The distributed file management system shown in drawing 11 is equipped with the networks 101, such as a Local Area Network which connects mutually two or more computers site A102 equipped with the client computer group which consists of two or more client computers, such as server computers, such as a personal computer and a workstation, and a personal computer, and a workstation, the computer site B1103 and the computer site C1104, the computer site A1102 and the computer site B1103, and the computer site C1104, and a Wide Area Network.

[0111] Here, the computer site A1102 is equipped with two or more server computers (only "the server computer A1105" is shown in drawing 11), such as a personal computer and a workstation, and the client computer group A108 which consists of client computer 1-n (108-1 - 108-n), such as a personal computer and a workstation. This computer site A1102 has connected two or more server computers (only "the server computer A1105" is shown in drawing 11) and client computer groups A108 in the internal networks 131, such as Ethernet, for example, has become the Internet domain.

[0112] Moreover, like the computer site A1102, the computer site B1103 was equipped with the client computer group B109 which consists of two or more server computers (only "the server computer B1106" is shown in drawing 11), and two or more client computers, and the computer site C1104 is equipped with the client computer group C110 which consists of two or more server computers (only "the server computer C1107" is shown in drawing 11), and two or more client computers. Furthermore, these computer sites B1103 and the computer site C1104 -- the computer site A1102 -- the same -- two or more server computers (in drawing 11, only the "server computer B1106" and the "server computer C1107" are shown), and the client computer groups B109 and the client computer groups C110 -- each -- it has connected in the internal network 132 and the internal network 133, for example, has become the Internet domain [0113] The storage 115, such as a hard disk with which the server computer A1105 records the partial file of a distributed file. The network interface 113 for connecting with the internal networks 131, such as Ethernet. The partial file management section 111 which controls the writing and read-out to the storage 115 which is recording the partial file. With the state Management Department 814 which supervises the load to storage 115, the remaining capacity of storage 115, and the load to a network interface 113, and holds the information about such loads and capacity. It is constituted by the partial file management section 111, the state

Management Department 814, and the distributed file management section 1112 connected to the network interface 113.

[0114] This state Management Department 814 has the external state Management Department 811 holding the external load information which notified load information to other server computers, and was notified from other server computers.

[0115] Moreover, the distributed file management section 1112 determined the partial file to which it is made to move based on the information on the access information for every partial file, and the load information external load information table 401 and 901, and is equipped with the distributed file move section 1131 which moves a partial file to other server computers.

[0116] The server computer B1106 and the server computer C1107 have the same composition as the server computer A1105. That is, the server computer B1106 is constituted by storage 120, a network interface 118, the partial file management section 116, the state Management Department 819 having the external state Management Department 812, and the distributed file management section 1117 equipped with the distributed file move section 1132. Moreover, the server computer C1107 is constituted by storage 125, a network interface 123, the partial file management section 121, the state Management Department 824 having the external state Management Department 813, and the distributed file management section 1122 equipped with the distributed file move section 1133.

[0117] The difference with the distributed file management system shown in the distributed file management system shown in drawing 11 and drawing 8 here is a point equipped with the distributed file move sections 1131, 1132, and 1133 which the distributed file-management sections 1112, 1117, and 1122 shown in drawing 11 determine the partial file to which it is made to move, and make move a partial file to other server computers from the information on the access information for every partial file, and the load information external load information table 401 and 901.

[0118] Drawing 12 shows an example of the access information table 1201 for every partial file managed at the state Management Department 814. In drawing 12, this access information table 1201 consists of a "partial file identification child" for discriminating a partial file, and an item of "the access information around unit time." Moreover, "the access information around unit time" consists of items of "the number of times of access" which indicates the number of times of access to a partial file to be the "accessing agency site identifier" which is the information on a site that the client computer which has accessed the partial file exists. It is continued by updating this access information table 1201 the state Management Department 814 for every unit time.

[0119] Drawing 13 shows an example of the partial file control table 1301 managed in the distributed file management section 1112. In drawing 13, the partial file control table 1301 consists of items of the "original address" which shows the "address" which shows the "partial file identification child" for discriminating a partial file, and the address of a partial file, and the address where the partial file was created first. In drawing 13, in the stage where the partial file was created, although the information which an "address" and an "original address" show is the same, the information on an "address" changes according to a partial file moving to other server computers. The partial file control table 1301 shown by drawing 13 is what added the item of an "original address" to the partial file control table 601 shown by drawing 6.

[0120] In the distributed file management system constituted as mentioned above, as the distributed file A, the distributed file B, and the distributed file C show by ***, after they are created by the server computer A1105, the processing which moves the partial file of each distributed file is explained in detail.

[0121] Drawing 14 shows an example of the contents of the partial file control table 1301 of the distributed file A created by the server computer A1105, the distributed file B, and the distributed file C. In drawing 14, drawing 14 (A) shows the partial file control table 1401 of the server computer A1105, (B) shows the partial file control table 1402 of the server computer B1106, and (C) shows the partial file control table 1403 of the server computer C1107. The address of each partial file where a partial file identification child is shown by A1, A2, A3, B1, C1, and C2, and the original address are shown in the partial file control table 1401 of drawing 14 (A).

The address of each partial file where a partial file identification child is shown by C1 and C2, and the original address are shown in the partial file control table 1402. The address of each partial file where a partial file identification child is shown by A2 and A3, and the original address are shown in the partial file control table 1403. Here, in drawing 14, the state before movement of each partial file is expressed. For this reason, in all partial files, the address and original address are in agreement.

[0122] The algorithm of the distributed file move section 1131 of the server computer A1105 in the case of movement of the partial file in the state which showed in drawing 14 of operation is explained.

[0123] Drawing 15 shows the algorithm of the distributed file move section 1131 of operation. First, the distributed file move section 1131 supervises the information on the load information table 401 (drawing 4) which the state Management Department 814 has managed at intervals of fixed time (Step 1501).

[0124] The distributed file move section 1131 will look for the "partial file identification child" of the partial file included in this detected storage with reference to the partial file control table 1301, if it detects that the "load" of a certain storage exceeded the predetermined threshold (for example, it is values, such as 80 etc.%, and the structure of a system etc. determines this value arbitrarily) set up beforehand (Step 1501). Discovered "the access information per unit time" of a "partial file identification child" is acquired from the access information table 1201. "the number of times of access" of "the access information per unit time" acquired here -- as compared with every partial file identification child --, the "partial file identification child" who is biggest "number of times of access" is chosen (Step 1502). That is, a move dimension partial file is chosen. For example, suppose that the partial file A1 (126a) was chosen as a move dimension partial file here.

[0125] Next, the distributed file move section 1131 chooses the server computer in which "remaining capacity" appears enough and a "load" has low storage from a predetermined value based on the external load information table 901 (drawing 9). And the distributed file move section 1131 checks whether a partial file can move to the selected server computer, and determines the server computer which can move (Step 1502). That is, a movement place server computer is chosen. For example, suppose that the storage section of the storage 125 shown by the storage identifier DiskID2 of the server computer C1107 was chosen here.

[0126] Next, a partial file is moved based on the information acquired by selection of this move dimension partial file, and selection of a movement place server computer (Step 1503).

[0127] In an above-mentioned example, by Step 1502, since the server computer C1107 is chosen as a move dimension partial file as the partial file A1 (126a) and a movement place server computer, the distributed file move section 1131 of the server computer A1105 of a moved material reads the partial file A1 (126a) from storage 115 through the partial file management section 111. This read partial file A1 (126a) is sent out through a network interface 113 and the internal network 131 in a network 101 with the information about the "original address" (drawing 13) of the partial file A1 (126a).

[0128] On the other hand, in the server computer C1107 of a movement place, the distributed file move section 1133 receives the information on the partial file A1 (126a) sent out from the server computer A1105 of a moved material, and its "original address" through the interior network 133 of network 101 shell, and a network interface 123. The partial file management section 121 writes this partial file A1 (126a) that received in storage 125. Moreover, the "original address" of the partial file A1 (126a) is registered into the partial file control table (drawing 14 (C)) of the partial file management section 121.

[0129] Then, the server computer C1107 of a movement place notifies that movement of the partial file A1 (126a) was completed to the server computer (the server computer A1105 with the server computer A1105 same in the case of this example, i.e., a moved material and original) shown in the server computer A1105 and an "original address" (drawing 13) of a moved material. By the server computer shown in the server computer A1105 and an "original address" of a moved material, the information on the partial file A1 (126a) registered into the partial file control table is rewritten.

[0130] Drawing 16 shows the state of the partial file control tables 1401, 1402, and 1403 shown in drawing 14 after the partial file A1 (126a) moves like an above-mentioned example. In drawing 16, drawing 16 (A) shows the partial file control table 1601 of the server computer A1105, (B) shows the partial file control table 1602 of the server computer B1106, and (C) shows the partial file control table 1603 of the server computer C1107. That is, each partial file control table 1401, 1601, 1602, and 1603 of drawing 16 (A) – (C) correspond to each partial file control tables 1401, 1402, and 1403 of drawing 14 (A) – (C), respectively. Here, the “address” of each partial file where a “partial file identification child” is shown by A1, A2, A3, B1, C1, and C2, and the “original address” are shown in the partial file control table 1601. The “address” of each partial file where a “partial file identification child” is shown by C1 and C2, and the “original address” are shown in the partial file control table 1602. The “address” of each partial file where a “partial file identification child” is shown by A1, A2, and A3, and the “original address” are shown in the partial file control table 1603. Here, the difference in the state of the partial file control tables 1401, 1402, and 1403 shown in the state and drawing 14 of the partial file control tables 1601, 1602, and 1603 shown in drawing 16 depends on the partial file A1 (126a) on having made it move to the server computer C1107 from the server computer A1105. Namely, the difference between drawing 16 and drawing 14 is set to the partial file control table 1401 of drawing 14 (A). The information registered as the “address” of the partial file A1 (126a) is “file://siteA/serveA/DisilD1/” in the partial file control table 1601 of drawing 16 (A). The “addresses” of the partial file A1 (126a) is “file://siteC/serveC/DiskID2/”, the point registered, and the point that the item of the partial file A1 is added in the partial file control table 1603 of drawing 16 (C).

[0131] Drawing 17 shows a partial file control table when the distributed file move section 1132 of the server computer B1106 moves the partial file C1 (126e) to the server computer C1107 and the distributed file move section 1133 of the server computer C1107 moves the partial file A1 (126a) to the server computer B1106 like the processing mentioned above further from the state which showed in this drawing 16.

[0132] In drawing 17, drawing 17 (A) shows the partial file control table 1701 of the server computer A1105, (B) shows the partial file control table 1702 of the server computer B1106, and (C) shows the partial file control table 1703 of the server computer C1107. The “address” of each partial file where a “partial file identification child” is shown by A1, A2, A3, B1, C1, and C2, and the “original address” are shown in the partial file control table 1701. The “address” of each partial file where a “partial file identification child” is shown by C1, C2, and A1, and the “original address” are shown in the partial file control table 1702. Moreover, the “address” of each partial file where a “partial file identification child” is shown by A2, A3, and C1, and the “original address” are shown in the partial file control table 1703.

[0133] Here, the difference in the state of the partial file control tables 1601, 1602, and 1603 shown in the state and drawing 16 of the partial file control tables 1701, 1702, and 1703 shown in drawing 17 depends having moved the partial file A1 (126a) to the server computer B1106 from the server computer C1107, and the partial file C1 (126e) having made it move to the server computer C1107 from the server computer B1106.

[0134] That is, corresponding to movement of the partial file A1 (126a), the item of the partial file A1 (126a) is added to the partial file control table 1702. Moreover, in the partial file control table 1703, the item (refer to drawing 16) of the partial file A1 (126a) is deleted. Furthermore, with reference to the “original address” of the partial file A1 (126a), the server computer B1106 told the server computer A1105 about movement, and has changed into “file://siteB/serveB/DiskID2/” the “address” of the partial file A1 (126a) registered into the partial file control table 1701 by this notice in the server computer A1105 shown in an “original address.”

[0135] Moreover, corresponding to movement of the partial file C1 (126e), the “address” of the partial file C1 (126e) registered into the partial file control table 1702 is changed into “file://siteC/serveC/DiskID3/” from “file://siteB/serveB/DisilD3/” (drawing 16 (B)).

Moreover, in the partial file control table 1703, the item of the partial file C1 (126e) is added.

[0136] Next, in the state which shows in drawing 17, in case the client computer 1 (108-1)

refers to the distributed file C, operation in case the content of reference is included in the partial file C1 (126e) is explained.

[0137] (1) The client computer 1 (108-1) requires reference of the distributed file C from the server computer A1105 which created the distributed file C. By the server computer A1105, it investigates which partial file is referred to with reference to the distributed file control table 1701 among the partial files C1 and C2 which constitute the distributed file C. Here, it considers as the partial file C1 (126e). Since the “address” of the partial file C1 (126e) is “file://siteB/serveB/DiskID3/”, the server computer A1105 checks whether the partial file C1 (126e) exists to the server computer B1106.

[0138] (2) The server computer B1106 investigates the distributed file control table 1702, and investigates the “address” of the partial file C1 (126e). Since the “address” of the partial file C1 (126e) is “file://siteC/serveC/DiskID3/”, the server computer B1106 checks whether the partial file C1 (126e) exists to the server computer C1107.

[0139] (3) The server computer C1107 investigates the distributed file control table 1703, and investigates the “address” of the partial file C1 (126e). Since the “address” of the partial file C1 (126e) is “file://siteC/serveC/DiskID3/”, it turns out that the partial file C1 (126e) exists in the server computer C1107.

[0140] (4) Notify that the server computer C1107 exists in the server computer B1106, and the partial file C1 (126e) exists in “file://siteC/serveC/DiskID3/”.

[0141] (5) In response to this notice, the server computer B1106 notifies that the partial file C1 (126e) exists in the server computer A1105 at “file://siteC/serveC/DiskID3/”.

[0142] (6) The server computer A1105 requires reference of the partial file C1 (126e) of the server computer C1107. Simultaneously with this demand, it directs to give the reference demand of the partial file C1 (126e) to the server computer C1107 directly to the client computer 1 (108-1) which required reference. Moreover, by the server computer A1105, the “address” of the partial file C1 (126e) is rewritten to “file://siteC/serveC/DiskID3/”.

[0143] As mentioned above, with the gestalt of this operation, the distributed file management sections 1112, 1117, and 1122 determine the partial file to which it is made to move based on each information on the access information load information table [external load information] 1201, 401, and 901 for every partial file. Moreover, concentration of the load to the storage of a specific server computer is avoidable by moving a partial file to other server computers by having the distributed file move sections 1131, 1132, and 1133 which move a partial file to other server computers.

[0144] In addition, with the gestalt of the 3rd operation mentioned above, it sets to Step 1501 of the algorithm (drawing 15) of the distributed file move section 1131 of operation. Instead of detecting that the “load” of each storage section of storage exceeded the predetermined value, You may make it detect that “remaining capacity” of each storage section of storage was less than values (however, this value is determined according to equipment or the structure of a system), such as a predetermined value [Mbytes], 10 [for example,] etc. The imbalance of the capacity of each storage section of storage is avoidable with this.

[0145] Moreover, in the above-mentioned step 1501, “use communication-band width of face” can detect the link beyond the predetermined value (however, this value is determined according to equipment or the structure of a system), for example, the value of 80 [%] of usable communication-band width of face, from the “load information” on a network 101 instead of detecting that the “load” of each storage section of storage exceeded the predetermined value. Moreover, in Step 1502, concentration of the load of a network is avoidable by choosing from the access information table 1201 the computer site which is raising the partial file which is raising the load of a network 101, and the load of a network. For example, when the “use communication-band width of face” (drawing 4) of the link of the computer site A1102 (sending out agency site) and the computer site B1103 (sending-out place site) exceeds a predetermined value, it exists in the server computer A1105, and the partial file used as the cause which raises a network load is moved to the server computer B1103. Thereby, the “use communication-band width of face” between the computer site A1102 and the computer site B1103 can be decreased.

[0148] Moreover, with the gestalt of the 3rd operation of a ****, although it checks whether the partial file of the server computer of a movement place is movable in Step 1502 and the partial file is moved in Step 1503, check processing of Step 1502 can be omitted by moving a partial file, without whether movement of a partial file at Step 1502 is possible in advance, and checking. When movement of a partial file is unacceptable by the server computer side of the movement place of a partial file at this time, the server computer of a movement place looks for the movement place for moving a partial file further, and should just move this partial file.

[0147] moreover -- although it is moving to the server computer of a movement place from the server computer of partial file move--origin in the above-mentioned step 1503 -- the move processing -- in addition, other partial files which can move to the server computer of a moved material out of the partial file in the server computer of a movement place -- choosing -- being concerned -- others -- you may make it move a partial file to the server computer of a moved material. This can protect that a partial file concentrates on one server computer, and the load to a file access can be mitigated more.

[0148] Moreover, in the above-mentioned step 1502, in case the server computer of the movement place of a partial file is chosen, it is good to choose the server computer in which a server computer list is beforehand set, has the remaining capacity of enough of each storage section of storage, and a load has low storage from a predetermined value out of the server computer under list. By this, the time spent on selection of the server computer of the movement place of a partial file can be shortened.

[0149] (Gestalt 4 of operation) Drawing 18 is a flow chart which shows other algorithms of the distributed file move sections 1131, 1132, and 1133 of distributed file management system of operation shown in drawing 11.

[0150] In drawing 18, the distributed file move sections 1131, 1132, and 1133 supervise the communication cost to each partial file at the predetermined intervals first (Step 1801). Here, as communication cost, it can consider as the communication time between the client computer which is referring to the partial file, and the server computer holding the partial file, for example. In drawing 11, communication cost of the client computer 1 (108-1) and the partial file A2 (126c) is taken as the communication time between the client computer 1 (108-1) which is referring to the partial file A2 (126c), and the server computer C1107 holding the partial file A2 (126c).

[0151] the distributed file move sections 1131, 1132, and 1133 exceeded the value predetermined in the communication cost to a partial file, for example, 1 etc. second etc., here (however, this value is determined according to equipment or the structure of a system) -- detecting (Step 1801) -- communication cost chooses as a partial file of partial file beyond predetermined value move--origin (Step 1802) Moreover, when two or more client computers have accessed to the partial file whose communication cost of this exceeded the predetermined value, it asks for the communication cost to each access, and these are added and it asks for sum total communication cost.

[0152] In case the server computer of a movement place is chosen, based on the external load information table 901, there is "remaining capacity" of enough of each storage section of storage, and the "load" to them chooses a server computer with low storage from a predetermined value. And as a result of transmitting above-mentioned sum total communication cost and moving a partial file to the selected server computer, it asks in order how communication cost changes, and the server computer which becomes the minimum communication cost is chosen (Step 1802). Or the distributed file move sections 1131, 1132, and 1133 have an initial entry between sites, expect communication cost from the initial entry, and you may make it choose the server computer which becomes the minimum communication cost (Step 1802).

[0153] Drawing 19 is drawing showing an example of an initial-entry table. In drawing 19, the initial-entry table 1901 has the item of "communication time" to show the communication cost from the "sending out agency site" which sends out a partial file, the "sending-out place site" where a partial file is sent out, and a sending out agency site to a sending-out place site. The "communication time" between a server computer and a client computer can be obtained from

this initial-entry table 1901 by to which site the server computer holding a partial file belongs to which site, and the client computer which refers to a partial file belongs. Based on this "communication time" and the sum total communication cost mentioned above, the server computer by which "communication time" serves as the minimum after movement of a partial file is chosen as a movement place of a partial file (Step 1802). When two or more client computers have accessed the partial file whose communication cost exceeded the predetermined value here, it is good to choose the server computer by which the "communication time" after movement of a partial file becomes the minimum communication cost.

[0154] For example, when the communication cost (henceforth "cost A2") of the client computer 1 (108-1) and the partial file A2 (126c) exceeds a predetermined value and the partial file A2 (126c) is moved to the server computer A1105, it asks the server computer A1105 what communication cost becomes, or asks for communication cost from the initial-entry table 1901. The result will make the server computer A1105 the candidate of a movement place server computer, if less than cost A2. This processing is performed also to other server computers, and the server computer by which communication cost becomes the minimum is looked for (Step 1802).

[0155] Finally, a partial file is moved based on the information acquired by selection of this move dimension partial file, and selection (Step 1802) of a movement place server computer (Step 1803).

[0156] With the gestalt of the 4th operation, the distributed file move sections 1131, 1132, and 1133 As mentioned above, each information on the load information external load information table [access information] 401, 901, and 1201, It is based on the initial entry between the server computer obtained from the initial-entry table 1901, and a client computer. It asks for the communication cost between the server computer by which the partial file under processing exists, and the client computer of the demand origin of processing. Since the partial file was moved to other server computers which become small [communication cost] when communication cost exceeded a predetermined value in addition to the effect which was mentioned above and which is acquired with the gestalt of the 3rd operation, the average of the access time to the partial file from a client computer can be shortened.

[0157] In addition, although communication time is mentioned as the example with the gestalt of the 4th operation of a **** as communication cost, it can also be made "delay" of communication time, "fluctuation (range of fluctuation) etc.", etc.

[0158] Moreover, although the partial file is moved with the gestalt of the 4th operation of a **** so that sum total communication cost may be made into the minimum when two or more client computers have accessed the same partial file, you may make it move a partial file so that average communication cost may be made into the minimum.

[0159] (Gestalt 5 of operation) Drawing 20 is the block diagram showing an example of the gestalt of operation of the 5th of the distributed file management system in this invention. In this drawing 20, the same sign is given to the same composition as drawing 8. The distributed file management system shown in drawing 20 is equipped with the networks 101, such as a Local Area Network which connects mutually two or more computers site A2002 equipped with the client computer group which consists of two or more client computers, such as server computers, such as a personal computer and a workstation, and a personal computer, and a workstation, the computer site B2003 and the computer site C2004, the computer site A2002 and the computer site B2003, and the computer site C2004, and a Wide Area Network.

[0160] Here, the computer site A2002 is equipped with two or more server computers (only "the server computer A2003" is shown in drawing 20), such as a personal computer and a workstation, and the client computer group A108 which consists of client computer 1-n (108-1 - 108-n), such as a personal computer and a workstation. This computer site A2002 has connected two or more server computers (only "the server computer A2003" is shown in drawing 20) and client computer groups A108 in the internal networks 131, such as Ethernet, for example, has become the Internet domain.

[0161] Moreover, like the computer site A2002, the computer site B2003 was equipped with the

client computer group B109 which consists of two or more server computers (only "the server computer B2006" is shown in drawing 20), and two or more client computers, and the computer site C2004 is equipped with the client computer group C110 which consists of two or more server computers (only "the server computer C2007" is shown in drawing 20), and two or more client computers. Furthermore, these computer sites B2003 and the computer site C2004 — the computer site A2002 — the same — two or more server computers (in drawing 20 , only the "server computer B2006" and the "server computer C2007" are shown), and the client computer groups B109 and the client computer groups C110 — each — it has connected in the internal network 132 and the internal network 133, for example, has become the Internet domain [0162] The storage 115, such as a hard disk with which the server computer A2005 records the partial file of a distributed file. The network interface 113 for connecting with the internal networks 131, such as Ethernet. The partial file management section 111 which controls the writing and read-out to the storage 115 which is recording the partial file. With the state Management Department 814 which supervises the load to storage 115, the remaining capacity of storage 115, and the load to a network interface 113, and holds the information about such loads and capacity. It is constituted by the partial file management section 111, the state Management Department 814, and the distributed file management section 2012 connected to the network interface 113.

[0163] This state Management Department 814 has the external state Management Department 811 holding the external load information which notified load information to other server computers, and was notified from other server computers.

[0164] Moreover, the distributed file management section 2012 determined the partial file to copy based on each information for every partial file obtained from the access information load information table [external load information] 1201 (drawing 12), 401 (drawing 4), and 901 (drawing 9), and is equipped with the distributed file copy section 2001 which copies the partial file concerned to other server computers.

[0165] The server computer B2006 and the server computer C2007 have the same composition as the server computer A2005. That is, the server computer B2006 is constituted by storage 120, a network interface 118, the partial file management section 118, the state Management Department 819 having the external state Management Department 812, and the distributed file management section 2017 equipped with the distributed file copy section 2032. Moreover, the server computer C2007 is constituted by storage 125, a network interface 123, the partial file management section 121, the state Management Department 824 having the external state Management Department 813, and the distributed file management section 2022 equipped with the distributed file copy section 2033.

[0166] The differences with the distributed file management system shown here in the distributed file management system shown in drawing 20 and drawing 8 are the distributed file copy sections 2031 and 2032 which determine the partial file which the distributed file management sections 2012, 2017, and 2022 shown in drawing 20 copy based on each information for every partial file obtained from the access information load information table [external load information] 1201, 401, and 901, and copy the partial file concerned to other server computers, and a point equipped with 2033 **.

[0167] Operation of the distributed file management system constituted as mentioned above is explained in detail taking the case of the case where a partial file is copied, after being created, as the distributed file A, the distributed file B, and the distributed file C show drawing 20 by the server computer A2005.

[0168] Drawing 21 is a flow chart which shows the algorithm of the distributed file copy section 2031 of the server computer A2005 of operation. In drawing 21 , the distributed file copy section 2031 supervises first the load information table 401 (drawing 4) which the state Management Department 814 has managed by the predetermined time interval (Step 2101).

[0169] as for the distributed file copy section 2031, the "load" of the arbitrary storage sections of storage 115 exceeded values, such as a predetermined value [%], 80 [for example,] etc. -- detecting (Step 2101) -- the partial file included in the storage section of the detected storage 115 is discovered with reference to the partial file control table 1301 (drawing 13) And the

access information on the discovered partial file is acquired from the access information table 1201. "The number of times of access" of the acquired access information is compared, and the partial file which is biggest "number of times of access" is chosen as a copy dimension partial file. Here, suppose that the partial file A1 (126a) was chosen. Next, based on the external load information table 901 (drawing 9), there is "remaining capacity" of enough of the storage section of storage, and the server computer in which a "load" has low storage from a predetermined value is chosen. And it checks whether a partial file can be copied to the selected server computer, and the server computer which can be copied is chosen as a copy place server computer (Step 2102).

[0170] Here, suppose that the storage section (storage identifier : DiskID2) of the storage 125 of the server computer C2007 was chosen.

[0171] Based on the information acquired by selection of this copy dimension partial file, and selection of a copy place server computer, a partial file is copied (Step 2103) and surveillance processing of Step 2101 is continued again.

[0172] Since the server computer C2007 was chosen as a copy dimension partial file at Step 2102 as the partial file A1 (126a) and a copy place server computer in the case of the above-mentioned example, here The distributed file copy section 2031 of the server computer A2005 of a copied material Through the partial file management section 111, the partial file A1 (126a) is read from storage 115, and this partial file A1 (126a) is sent out to a network 101 through a network interface 113. Moreover, the information about the "original address" (drawing 13) of the partial file A1 (126a) is also sent out simultaneously.

[0173] On the other hand, by the server computer C2007 of a copy place, the distributed file copy section 2033 receives the partial file A1 (126a) sent out from the server computer A2005 of a copied material through a network interface 123. And it writes in the predetermined storage section of storage 125 through the partial file management section 121. Moreover, the "original address" of the partial file A1 (126a) is received, and it registers with the partial file control table 1403. Then, the server computer C2007 of a copy place notifies that the copy of the partial file A1 (126a) was completed to the server computer (in the case of this example, an "original address" is the storage section of the storage 115 of the server computer A2005) shown in the server computer A2005 and an "original address" of a copied material. By the server computer (both server computer A2005) shown in the server computer and an "original address" of a copied material, the information on the partial file A1 (126a) of the partial file control table 1401 is rewritten.

[0174] Drawing 22 is drawing showing the partial file control table of the server computer after copy processing. The partial file control tables 1401, 1402, and 1403 shown in drawing 14 change to the state of the partial file control tables 2201, 2202, and 2203 shown in drawing 22 as a result of the copy of the above-mentioned partial file A1 (126a). That is, drawing 22 (A) shows the partial file control table 2201 of the server computer A2005, (B) shows the partial file control table 2202 of the server computer B2006, and (C) shows the partial file control table 2203 of the server computer C2007. Moreover, the "address" of each partial file where a "partial file identification child" is shown by A1, A2, A3, B1, C1, and C2, and the "original address" are shown in the partial file control table 2201. Moreover, the "address" of each partial file where a "partial file identification child" is shown by C1 and C2, and the "original address" are shown in the partial file control table 2202. Moreover, the "address" of each partial file where a "partial file identification child" is shown by A1, A2, and A3, and the "original address" are shown in the partial file control table 2203. The difference between the state of a partial file control table shown in drawing 14 and the state of a partial file control table shown in drawing 22 is a difference by having copied the partial file A1 (126a) to the server computer C2007 from the server computer A2005. Namely, it sets to the partial file control table 1401 of drawing 14 (A), As opposed to the "address" of the partial file A1 (126a) being "file://siteA/serverA/DiskID1/" In the partial file control table 2201 of drawing 22 (A) The point that "address" ** is "file://siteA/serverA/DiskID1/" and "file://siteC/serverC/DiskID2/(address of a copy)" is different. [of the partial file A1 (126a)] Furthermore, in the partial file control table 2103 of drawing 22 (C), the item of the partial file A1 (126a) is added.

[0175] Moreover, from the state of drawing 22, the distributed file copy section 2302 of the server computer B2006 can copy the partial file C1 (126e) to the server computer C2007, and the distributed file copy section 2303 of the server computer C2007 can also copy the partial file A1 (126a) to the server computer B2006.

[0176] Drawing 23 is drawing showing the partial file control table in the state where the partial file was copied further, from the state of drawing 22. Drawing 23 (A) shows the partial file control table 2301 of the server computer A2005. (B) shows the partial file control table 2302 of the server computer B2006, and (C) shows the partial file control table 2303 of the server computer C2007. The "address" of each partial file of A1, A2, A3, B1, C1, and C2 and the "original address" are shown for the "partial file identification child" in the partial file control table 2301 of drawing 23 (A). The "address" of each partial file of C1, C2, and A1 and the "original address" are shown for the "partial file identification child" in the partial file control table 2302 of drawing 23 (B). Moreover, the "address" of each partial file of A1, A2, A3, and C1 and the "original address" are shown for the "partial file identification child" in the partial file control table 2303 of drawing 23 (C). The difference between the state of the partial file control table of drawing 22 and the state of the partial file control table of drawing 23 is because the partial file C1 (126e) was copied [having copied the partial file A1 (126a) to the server computer B2006 from the server computer C2007, and] to the server computer C2007 from the server computer B2006.

[0177] That is, corresponding to the copy of the partial file A1 (126a), the item of the partial file A1 (126a) is added in the partial file control table 2302. Moreover, the server computer A2005 of an "original address" has added "file://siteB/serveB/DiskID2/" to the "address" of the partial file A1 (126a) in the partial file control table 2301 because the server computer B2006 tells the server computer A2005 about a copy with reference to the "original address" of the partial file A1 (126a). Moreover, corresponding to the copy of the partial file C1 (126e), the item of the partial file C1 (126e) is added in the partial file control table 2303. Furthermore, in the partial file control table 2302, the "address" of the partial file C1 (126e) is "file://siteB/serveB/DiskID3/" and "file://siteC/serveC/DiskID3/(address of a copy)".

[0178] In the state which shows in drawing 23, in case the client computer 1 (108-1) refers to the distributed file C, operation in case the content of reference is included in the partial file C1 (126e) is explained.

[0179] (1) The client computer 1 (108-1) requires reference of the distributed file C from the server computer A2005 which created the distributed file C. By the server computer A2005, it investigates which partial file is referred to with reference to the distributed file control table 2301 among the partial files C1 and C2 which constitute the distributed file C, and recognizes that it is reference of the partial file C1 (126e). Since the "address" of the partial file C1 (126e) is "file://siteB/serveB/DiskID3/" in the distributed file control table 2301, the server computer A2005 checks whether the partial file C1 (126e) exists to the server computer B2006.

[0180] (2) The server computer B2006 investigates the distributed file control table 2302, and investigates the address of the partial file C1 (126e). Since the "addresses" of the partial file C1 (126e) is "file://siteB/serveB/DiskID3/" and "file://siteC/serveC/DiskID3/" in the distributed file control table 2302, the server computer B2006 chooses the low server computer of a load from the load information table 401 and the external load information table 901 among the server computer B2006 and the server computer C2007. Here, when the server computer C2007 is chosen, the server computer B2006 checks whether the partial file C1 (126e) exists to the server computer C2007.

[0181] (3) The server computer C2007 investigates the distributed file control table 2303, and checks the "address" of the partial file C1 (126e). Since the "address" of the partial file C1 (126e) is "file://siteC/serveC/DiskID3/" in the distributed file control table 2303, it turns out that the partial file C1 (126e) exists in the server computer C2007.

[0182] (4) The server computer C2007 notifies that the partial file C1 (126e) exists at "file://siteC/serveC/DiskID3/" to the server computer B2006.

[0183] (5) The server computer B2006 notifies that the partial file C1 (126e) exists at "file://siteC/serveC/DiskID3/" to the server computer A2005.

[0184] (6) The server computer A2005 requires reference of the partial file C1 (126e) of the server computer C2007. Simultaneously with this demand, it is directed that the client computer 1 (108-1) gives the reference demand of the partial file C1 (126e) to the server computer C2007 directly to the client computer 1 (108-1) which required reference.

[0185] With the gestalt of this operation, as mentioned above, the distributed file management sections 112, 117, and 122 it is based on each information for every partial file obtained from the access information load information table [external load information] 1201, 401, and 901. When the partial file to copy is determined and the distributed file copy sections 2301, 2302, and 2303 copy a partial file to other server computers, concentration of the load to the storage of a specific server computer is avoidable.

[0186] In addition, with the gestalt of the 5th operation, you may decide to replace with detecting the thing which were mentioned above and for which the "load" of each storage section of storage exceeded the predetermined value, and to detect the link "whose use communication-band width of face" of the network load information table 403 exceeded the predetermined value in Step 2101 of the algorithm (drawing 21) of the distributed file copy section 2031 of operation. Moreover, you may make it choose from the access information table 1201 the partial file which is raising the network load, and the computer site which is raising the network load in Step 2102. Concentration of a network load is avoidable with this. For example, when the "use communication-band width of face" of the link of the computer site A2002 (sending out agency site) and the computer site B2003 (sending-out place site) exceeds a predetermined value, it exists in the server computer A2005, and the partial file used as the cause which raises a network load is copied to the server computer B2003. Thereby, the "use communication-band width of face" between the computer site A2002 and the computer site B2003 decreases.

[0187] Drawing 24 is a flow chart which shows other algorithms of the distributed file copy section 2031 of the server computer A2005 of operation. In drawing 24, the distributed file copy section 2031 supervises the communication cost to a partial file at the predetermined intervals first (Step 2401).

[0188] Here, it can consider as the communication time between the client computer which is referring to for example, the partial file, and the server computer holding the partial file at communication cost. For example, what is necessary is just to make it into the communication time between the client computer 1 (108-1) which is referring to the partial file A2 (126c), and the server computer C2007 holding the partial file A2 (126c) in the case of the communication cost of the client computer 1 (108-1) and the partial file A2 (126c).

[0189] Next, the distributed file copy section 2031 chooses the partial file to which this communication cost exceeded the predetermined value as a copy dimension partial file, when the communication cost to a partial file detects having exceeded the predetermined value (Step 2401). When two or more client computers have accessed the partial file whose communication cost exceeded the predetermined value, it asks for each communication cost for every access, and these are added to it and it is asked for sum total communication cost. When choosing the server computer of a copy place, as a result of there having been "remaining capacity" of enough of the storage section of storage, and choosing the server computer in which a "load" has low storage, transmitting sum total communication cost to the selected server computer and copying a partial file on the other hand based on the external load information table 901, it is asked in order how communication cost changes. And the server computer which becomes the minimum communication cost is chosen as a copy place server computer. Or as an initial entry between sites (between a server computer and client computers), the distributed file copy sections 2031, 2032, and 2033 have the initial-entry table 1901 shown in drawing 19, expect communication cost (communication time) from the information, and you may make it choose the server computer which becomes the minimum communication cost. The "communication time" between a server computer and a client computer can be obtained using the information on the initial-entry table 1901 of drawing 19, and the information on to which site a server computer belongs to which site and a client computer belongs. And it is good to choose the server computer by which the "communication time" serves as the minimum as a copy place of a

partial file. When two or more client computers have accessed the partial file whose communication cost exceeded the predetermined value, the server computer which becomes the minimum sum total communication cost is chosen as a copy place server computer (Step 2402). [0190] For example, when the communication cost (cost A2) of the client computer 1 (108-1) and the partial file A2 exceeds a predetermined value and the partial file A2 is copied to the server computer A105, it is asked to the server computer A105 what communication cost becomes. Or it asks for cost from an initial entry 1801. The result will make the server computer A105 the candidate of a copy place, if less than cost A2. This processing is repeated and the server computer by which communication cost becomes the minimum is looked for. Based on the information acquired by selection of this copy dimension partial file, and selection of a copy place server computer, a partial file is copied (Step 2403) and surveillance processing of Step 2401 is continued again.

[0191] Thus, the average of the access time to the partial file from a client computer can be shortened by changing Step 2101 and Step 2102 of drawing 21 into Step 2401 and Step 2402 of drawing 24. Moreover, although "communication time" is mentioned as the example as communication cost above, "delay" of communication time, "fluctuation (range of fluctuation)" etc., etc. are sufficient.

[0192] Moreover, in Step 2102 of drawing 21, and Step 2402 of drawing 24, although it checks whether the copy of a partial file is possible to the server computer of a copy place and the partial file is copied to it in Step 2103 and Step 2403, check processing of Step 2102 and Step 2402 can be omitted by copying a partial file, without checking whether the copy of a partial file is possible to a server computer at Step 2102 and Step 2402. When the copy of a partial file cannot be accepted by the server computer side of a copy place in Step 2103 and Step 2403 at this time [whether the server computer of a copy place copies a partial file in search of the copy place for copying a partial file further, and] Or you may make it notify having canceled the partial file transmitted to the copy and having canceled the partial file for a copy to the server computer of a copied material.

[0193] Moreover, although the partial file is only copied to the server computer of a copy place from the server computer of a copied material in Step 2103 of drawing 21, and Step 2403 of drawing 24 with the gestalt of the 5th operation of a **** In addition to the copy processing, the partial file which may make it move to the server computer of a copied material is chosen from the partial files in the server computer of a copy place, and you may make it move to the server computer of the partial file move-origin. By this, a partial file cannot concentrate on one server computer, but concentration of the load to the storage of a specific server computer can be avoided.

[0194] Moreover, in Step 2102 of drawing 21, in case the server computer of a copy place is chosen, the server computer list is set up beforehand, there is remaining capacity of enough of the storage section of storage out of the server computer under this server computer list, and the server computer in which a load has low storage can be chosen. The time for server selection can be shortened now by this.

[0195] (Gestalt 6 of operation) Drawing 25 is a flow chart which shows other algorithms of the distributed file copy section 2031 of the server computer A2005 in the distributed file management system shown in drawing 20 of operation. In the gestalt of this operation, it has the same composition as the gestalt of the 5th operation mentioned above, and if operation of the distributed file copy section is removed, same operation is performed. Hereafter, the algorithm of the distributed file copy section 2031 of operation is explained.

[0196] In drawing 25, the distributed file copy section 2031 supervises first the load information table 401 (drawing 4) which the state Management Department 814 has managed by the predetermined time interval (Step 2501).

[0197] As for the distributed file copy section 2031, the "load" of the arbitrary storage sections of storage 115 exceeded values, such as a predetermined value [X], 80 [for example,] etc., -- detecting (Step 2501) -- the partial file included in the detected storage is discovered with reference to the partial file control table 1301 (drawing 13) And "the access information per unit time" on the discovered partial file is acquired from the access information table 1201. "The

number of times of access" of the acquired "the access information per unit time" is compared, and the partial file which is biggest "number of times of access" is chosen as a copy dimension partial file (Step 2502). Here, suppose that the partial file A1 was chosen, for example,

[0198] Next, based on the external load information table 901 (drawing 9), there is "remaining capacity" of enough of the storage section of storage, and two or more server computers in which a "load" has low storage from a predetermined value are chosen. And it checks whether a partial file can be copied to the selected server computer, and the server computer which can be copied is determined as a copy place server computer (Step 2502).

[0199] Here, suppose that the storage 120 which has the storage section shown by the storage identifier DiskID1 of the server computer B108, and the storage 125 which has the storage section shown by the storage identifier DiskID2 of the server computer C107 were chosen.

[0200] Next, based on the information on the copy dimension partial file obtained at the above-mentioned step 2502, and a copy place server computer, a partial file is copied (Step 2503) and surveillance processing of Step 2501 is continued.

[0201] In an above-mentioned example, since the server computer B2006 and the server computer C2007 were chosen as the partial file A1 (126a) and a copy place computer as a copy dimension partial file at Step 2502. The distributed file copy section 2031 of the server computer A2005 of a copied material minds the partial file management section 111. The partial file A1 (126a) is read from storage 115, and it transmits to a network 101 by multicasting through a network interface 113. Moreover, the information about the "original address" (drawing 13) of the partial file A1 (126a) is also transmitted simultaneously.

[0202] On the other hand, by the server computer B2006 and computer C2007 of a copy place, the distributed file copy sections 2032 and 2033 receive through network interfaces 118 and 123, and write the partial file A1 (126a) transmitted from the server computer A2005 of a copied material in storage 120 and 125 through the partial file management sections 116 and 121.

Moreover, the "original address" (drawing 13) of the partial file A1 (126a) is received, and it registers with each partial file control table 1402 and 1403 (drawing 14). Then, the server computer B2006 and the server computer C2007 of a copy place notify that the copy of the partial file A1 (126a) completed them to both and this server computer A2005 in the server computer shown in the server computer and an "original address" (drawing 13) of a copied material, respectively, i.e., this case, since both were the server computers A2005. The information on the partial file A1 (126a) of the partial file control table 1401 (drawing 14) is rewritten by the server computer A2005 shown in the server computer and an "original address" (drawing 13) of a copied material, i.e., a server computer.

[0203] As mentioned above, with the gestalt of this operation, since the candidate of the server computer of two or more copy places is chosen and a partial file is simultaneously copied to two or more selected server computers by multicasting communication in case the distributed file copy sections 2031, 2032, and 2033 copy a partial file to other server computers, the traffic in the case of the copy of a partial file is reducible.

[0204] (Gestalt 7 of operation) Drawing 26 is the block diagram showing an example of the gestalt of other operations of the distributed file management system in this invention. Here, in drawing 26, the same sign is given to the thing of the same composition as drawing 1. It set to drawing 26 and this distributed file management system is equipped with the networks 101, such as a Local Area Network which connects mutually two or more computers site A2602 equipped with the client computer group which consists of two or more client computers, such as server computers, such as a personal computer and a workstation, and a personal computer, and a workstation, the computer site B2603 and the computer site C2604, the computer site A2602 and the computer site B2603, and the computer site C2604, and a Wide Area Network.

[0205] Here, the computer site A2602 is equipped with two or more server computers (only "the server computer A2605" is shown in drawing 26), such as a personal computer and a workstation, and the client computer group A108 which consists of client computer 1-n (108-1 - 108-n), such as a personal computer and a workstation. This computer site A2602 has connected two or more server computers (only "the server computer A2605" is shown in drawing 26) and client computer groups A108 in the internal networks 131, such as Ethernet, for

example, has become the Internet domain.

[0206] Moreover, like the computer site A2602, the computer site B2603 was equipped with the client computer group B2609 which consists of two or more server computers (only "the server computer B2606" is shown in drawing 26), and two or more client computers, and the computer site C2604 is equipped with the client computer group C2610 which consists of two or more server computers (only "the server computer C2607" is shown in drawing 26), and two or more client computers, furthermore, these computer sites B2603 and the computer site C2604 -- the computer site A2602 -- the same -- two or more server computers (in drawing 26, only the "server computer B2606" and the "server computer C2607" are shown), and the client computer groups B109 and the client computer groups C110 -- each -- it has connected in the internal network 132 and the internal network 133, for example, has become the Internet domain [0207] The storage 115, such as a hard disk with which the server computer A2605 records the partial file of a distributed file. The network interface 113 for connecting with the internal networks 131, such as Ethernet The partial file management section 111 which controls the writing and read-out to the storage 115 which is recording the partial file. With the state Management Department 114 which supervises the load to storage 115, the remaining capacity of storage 115, and the load to a network interface 113, and holds the information about such loads and capacity it is constituted by the partial file management section 111, the state Management Department 114, and the distributed file management section 2612 connected to the network interface 113.

[0208] This distributed file management section 2612 directs writing and read-out of a partial file in the partial file management section 111. Moreover, when creating a distributed file, based on the information acquired from the state Management Department 114, the distributed file management section 2612 divides a distributed file into two or more partial files, and determines the server computer which arranges each partial file (record). Moreover, in referring to or updating the distributed file created before, it detects the server (recorded) computer by which the partial file of the corresponding distributed file exists. Here, the distributed file management section 2612 is equipped with the partial file size determination section 2631 which determines the size of the partial file at the time of dividing a distributed file to a partial file according to the kind of data recorded on information or a distributed file from a client computer.

[0209] The server computer B2606 and the server computer C2607 have the same composition as the server computer A2605. That is, the server computer B2606 is constituted by storage 120, a network interface 118, the partial file management section 116, the state Management Department 119, and the distributed file management section 2617. Moreover, the server computer C2607 is constituted by storage 125, a network interface 123, the partial file management section 121, the state Management Department 124, and the distributed file management section 2622. Moreover, the distributed file management sections 2617 and 2622 are equipped with the partial file size determination sections 2632 and 2633, respectively.

[0210] The difference with the distributed file management system shown in the distributed file management system shown in drawing 26 and drawing here is a point equipped with the partial file size determination sections 2631, 2632, and 2633 which determine the size of the partial file at the time of dividing a distributed file to a partial file according to the kind of data recorded on the distributed file management sections 2612, 2617, and 2622 by information or a distributed file from a client computer in the distributed file management system of drawing 26.

[0211] Operation of the partial file size determination sections 2631, 2632, and 2633 of the distributed file management system constituted as mentioned above is explained below.

[0212] In drawing 3, when the client computer 1 (108-1) performs the creation demand of the distributed file A to the server computer A2605, the partial file size determination section 2631 of the distributed file management section 2612 determines the size when assigning a portion of a portion by processing of Step 302. It is good to make it decide with the directions from the kinds (for example, M-JPEG, MPEG1, MPEG 2, etc.) and the client computer 1 (108-1) of data which the partial file size determination section 2631 records on a distributed file to determine the size of a partial file in the case.

[0213] As mentioned above, in the distributed file management system of the gestalt of this

operation, the partial file size determination sections 2631, 2632, and 2633 of the distributed file management sections 2612, 2617, and 2622 can change suitably the size of the partial file which constitutes a distributed file by determining the size of the partial file at the time of dividing a distributed file to a partial file according to the kind of data recorded on information and a distributed file from a client computer etc. Thereby, it can prevent dividing into two or more partial files, a logic target, the data which have relation in content, for example, the data for one picture etc., etc.

[0214] (Gestalt 8 of operation) Next, the distributed file management section and the state Management Department which two or more server computers mentioned above have are summarized to one server computer, and the case where it manages intensively by this server computer is explained.

[0215] Drawing 27 shows an example of the composition of the distributed managerial system at the time of summarizing the distributed file management section and the state Management Department to one server computer in the distributed file management system in this invention shown in the gestalt of the 1st operation. In drawing 27, the same sign is given to the thing of the same composition as drawing 1.

[0216] The distributed file management system shown in drawing 27 Two or more computer sites A2702, the computer site B2703 equipped with the client computer group which consists of two or more client computers, such as server computers, such as a personal computer and a workstation, and a personal computer, and a workstation And the computer site X2710 equipped with the management server computer X2711 which manages intensively the distributed file arranged on the computer site C2704 and other server computers, It has the networks 101, such as a Local Area Network which connects the computer site A2702, the computer site B2703, the computer site C2704, and the computer site X2710 mutually, and a Wide Area Network.

[0217] Here, the computer site A2702 is equipped with two or more server computers (only "the server computer A2705" is shown in drawing 27), such as a personal computer and a workstation, and the client computer group A108 which consists of client computer 1-n (108-1 - 108-n), such as a personal computer and a workstation. This computer site A2702 has connected two or more server computers (only "the server computer A2705" is shown in drawing 27) and client computer groups A108 in the internal networks 131, such as Ethernet, for example, has become the Internet domain.

[0218] Moreover, like the computer site A2702, the computer site B2703 was equipped with the client computer group B109 which consists of two or more server computers (only "the server computer B2706" is shown in drawing 27), and two or more client computers, and the computer site C2704 is equipped with the client computer group C110 which consists of two or more server computers (only "the server computer C2707" is shown in drawing 27), and two or more client computers, furthermore, these computer sites B2703 and the computer site C2704 -- the computer site A2702 -- the same -- two or more server computers (in drawing 27, only the "server computer B2706" and the "server computer C2707" are shown), and the client computer groups B109 and the client computer groups C110 -- each -- it has connected in the internal network 132 and the internal network 133, for example, has become the Internet domain [0219] The server computer A2705 is constituted by the storage 115, such as a hard disk which records the partial file of a distributed file, the network interface 113 for connecting with the internal networks 131, such as Ethernet, and the partial file management section 111 that controls the writing and read-out to the storage 115 which is recording the partial file.

[0220] The server computer B2706 and the server computer C2707 have the same composition as the server computer A2705. That is, the server computer B2706 is constituted by storage 120, a network interface 118, and the partial file management section 116. Moreover, the server computer C2707 is constituted by storage 125, a network interface 123, and the partial file management section 121.

[0221] The network interface 2713 for connecting the management server computer X2711 to the internal networks 134, such as Ethernet With the state Management Department 2714 which supervises the load of the storage of each server computer, and the remaining capacity and the load of a network interface, and holds the information about a load Point to writing and read-out

of a partial file in the partial file management sections 111, 116, and 121 of each server computer, or in case a distributed file is created, a distributed file is divided into two or more partial files based on the information from the state Management Department 2714. In creating a partial file and referring to or updating a distributed file by determining the server computer which arranges each partial file it is constituted by the distributed file management section 2712 which detects the server computer which the partial file of the distributed file concerned referred to or updated ****, and performs the reference or updating of a partial file.

[0222] In drawing 27, the state after each distributed files A, B, and C were created is shown. That is, the partial file A1 (126a) of the distributed file A and the partial file B1 (126b) of the distributed file B are recorded on the storage 115 of the server computer A2705. Moreover, the partial file C1 (126c) of the distributed file C and the partial file C2 (126f) of the distributed file C are recorded on the storage 120 of the server computer B2706. Moreover, the partial file A2 (126e) of the distributed file A and the partial file A3 (126d) of the distributed file A are recorded on the storage 125 of the server computer C2707.

[0223] Next, operation of the distributed file management system constituted as mentioned above is explained. The creation demand of the distributed file A is published to the server computer A2705 from the client computer 1 (108-1) of the client computer group A108 below, and distributed processing in case the partial files A1-A3 as shown in drawing 27 are created is made into an example, and is explained. Here, the storage 115, 120, and 125 shown in drawing 27 shall have two or more storage sections or storage regions (only henceforth the "storage section"), respectively. Two or more of these storage sections may be one record medium physically, and may be two or more record media.

[0224] In drawing 27, the creation demand of the distributed file A is first published by the management server computer X2711 of the computer site X2710 from the client computer 1 (108-1). The creation demand of this distributed file A is received by the distributed file management section 2712 through the internal network 131 of the computer site A2702, a network 101, the internal network 134 of the computer site X2710, and the network interface 2713 of the management server computer A2711.

[0225] Drawing 28 is a flow chart which shows the algorithm of the distributed file management section 2712 at the time of receiving the creation demand of a distributed file of operation. Hereafter, detailed operation of the distributed file management section 2712 is explained using drawing 27 and drawing 28.

[0226] Refer to the load information which the state Management Department 2714 has managed for the distributed file management section 2712 first in drawing 28 (Step 2801).

[0227] At the state Management Department 2714, a load information table 401 like drawing 4 is managed, for example. In drawing 4, although the information about the server computer A2705 is shown, at the state Management Department 2714, the load information table 401 as shown in drawing 4 is prepared and managed for every server computer. The load information table 401 consists of a storage load information table 402 and a network load information table 403 in drawing 4. The storage load information table 402 consists of information on the "storage identifier" for discriminating each storage section of the storage connected to the server computer, the "load" of each storage section of storage, and the "remaining capacity" of each storage section of storage. The "load" of each storage section of storage is displayed by what [%] is used among the maximum transfer rates of the storage section. The network load information table 403 means towards which site the data transmitted through the network interface of each server computer are transmitted using the bandwidth of how much, and whether it has received by sending TEFA which has received from which site using the bandwidth of how much. Moreover, a "sending out agency site" shows the computer site of the sending-out origin of data, a "sending-out place site" shows the computer site of the sending-out place of data, and the communication-band width of face for which "use bandwidth" is used between the computer site of a sending out agency and the computer site of a sending-out place is shown.

[0228] For example, when creating the distributed file A, about the storage section the storage identifier of the storage 115 of the server computer A2705 is indicated to be by DiskID 1, the

"load" is 20 [%] and the distributed file management section 2712 can acquire the information that the "remaining capacity" is 10 [Mbytes].

[0229] Next, the distributed file management section 2712 is based on the "storage load information" acquired from the state Management Department 2714. From each storage sections of the storage connected to each server computer, the value of "remaining capacity" is larger than a predetermined value, and the value of a "load" chooses a predetermined value, for example, the storage which has the low storage section from 80 [%] (this value is determined according to the composition of a system or other equipments). And a partial file is assigned to the storage section of this storage in order. At this time, it is good to make size of a partial file into the same fixed length by all server computers. In quota processing of this partial file, it detects whether all partial files were able to be assigned (Step 2802).

[0230] Here, when creating the distributed file A, in drawing 27, the partial file A1 (126a) will be assigned to the server computer A2705, and the partial file A2 (126c) and the partial file A3 (126d) will be assigned to the server computer C2704.

[0231] When all partial files are able to be assigned, the distributed file management section 2712 registers the information for managing a distributed file into the distributed file control table 501 of drawing 5 and the partial file control table 601 of drawing 6 which were mentioned above (Step 2803).

[0232] In the above-mentioned example, as shown in drawing 5, the distributed file A consists of a partial file A1 (126a), a partial file A2 (126c), and a partial file A3 (126d).

[0233] Moreover, in drawing 6, the address of the partial file A1 (126a) is "file:///siteA/serverA/DiskID1/(storage identifier DiskID1 of the server computer A105 of the computer site A102)." The address of the partial file A2 (126c) is "file:///siteC/serverC/DiskID2/(storage identifier DiskID2 of the server computer C107 of the computer site C104)." The address of the partial file A3 (126d) expresses that it is "file:///siteC/serverC/DiskID2/(storage identifier DiskID2 of the server computer C107 of the computer site C104)."

[0234] Next, the distributed file management section 2712 gives the creation demand of a partial file to the partial file management section of the server computer which creates in order to create the partial file registered at Step 2803 on each corresponding server computer.

Simultaneously with this creation demand, it directs to transmit data to the server computer which creates a direct partial file from the client computer 1 (108-1) to the client computer 1 (108-1) which performed the creation demand of a distributed file. By the server computer of which creation of a partial file was required, the data from the client computer 1 (108-1) are written in each storage section of storage by the partial file management section. The distributed file management section 2712 repeats this processing until creation of all partial files finishes (Step 2804).

[0235] Here, in creation of the partial file A1 (126a) of the distributed file A, the partial file management section 111 of the server computer A2705 writes the data from the client computer 1 (108-1) in the predetermined storage section of storage 115. Moreover, the partial file management section 121 of the server computer C2707 writes the data from the client computer 1 (108-1) in the predetermined storage section of storage 125, and creates the partial file A2 (126c) and the partial file A3 (126d).

[0236] On the other hand, at Step 2802, when all partial files are not able to be assigned, the distributed file management section 2712 notifies what creation processing of a distributed file went wrong to the client computer 1 (108-1) which performed the creation demand of a distributed file (Step 2805).

[0237] Next, the case where reference of a distributed file or the demand (henceforth "reference/updating demand") of updating is published from a client computer to a management server computer is explained. Moreover, the following explanation is described by making into an example the case where reference/updating demand of the distributed file A is published to the management server computer X2711 from the client computer 1 (108-1) in the client computer group A108.

[0238] First, reference/updating demand to the distributed file A published by the client computer 1 (108-1) is received by the distributed file management section 2712 through a

network interface 2713 in the management server computer X2711.
 [0239] Drawing 29 is a flow chart which shows an algorithm of operation when the distributed file management section receives reference/updating demand of a distributed file. Hereafter, operation of the distributed file management section 2712 is explained using drawing 29.
 [0240] First, the distributed file management section 2712 accepts reference/updating demand of the distributed file A from the client computer 1 (108-1), and it asks for the address of the partial file updated or referred to and its partial file from the distributed file control table 501 (drawing 5) and the partial file control table 601 (drawing 6) (Step 2901).
 [0241] Here, in reference/updating demand of the distributed file A, the distributed file control table 501 (drawing 5) shows that the distributed file A is constituted by the partial file A1 (126a), the partial file A2 (126c), and the partial file A3 (126d). By the partial file control table 601 (drawing 6), moreover, the partial file A1 (126a) it exists in the storage section of the storage 125 shown by "f1e://siteA/ServerA/DiskID1/", the partial file A2 (126c) it exists in the storage section of the storage 125 shown by "f1e://siteC/ServerC/DiskID2/", the partial file A3 (126d) it turns out that it exists in the storage section of the storage 125 shown by "f1e://siteC/ServerC/DiskID2/".

[0242] The distributed file management section 2712 performs reference/updating demand of the partial file concerned to the server computer holding the partial file corresponding to the reference/updating demand from the client computer 1 (108-1). It is directed that the distributed file management section 2712 gives reference/updating demand to the server computer by which the partial file to which the client computer 1 (108-1) performs reference or updating exists to the client computer 1 (108-1) directly simultaneously with this demand. The partial file management section of each server computer writes in read-out (reference) of the partial file which exists in storage, or the partial file to storage based on reference/updating demand from the client computer 1 (108-1) according to the demand from the distributed file management section 2712 (Step 2902). (updating)
 [0243] When it is the distributed file A here, in the partial file A1 (126a), the partial file A2 (126c) exists in the server computer C2707 of the computer site C2704, and the partial file A3 (126d) exists in the server computer A2705 of the computer site A2702 at the server computer C2707 of the computer site C2704. Therefore, processing of reference/updating demand to the partial file A1 (126a) is directly performed according to the demand from the distributed file management section 2712 between the client computer 1 (108-1) and the partial file management section 111. On the other hand, processing of reference/updating demand to the partial file A2 (126c) and the partial file A3 (126d) is directly performed according to the demand from the distributed file management section 2712 between the client computer 1 (108-1) and the server computer C2707.

[0244] As mentioned above, since management of a distributed file and management of the state of a system are performed [according to the distributed file management system of the gestalt of this operation] intensively in addition to the effect shown in the gestalt of the 1st operation, it is not necessary to have two or more duplicate Management Department, and simple [of the system configuration] can be carried out, and mitigation of cost can be aimed at.
 [0245] In addition -- although explained as composition which concentrated the Management Department of distributed file management system which showed with the gestalt of the 1st operation on one management server computer in the distributed file management system of drawing 27 mentioned above -- the 2- it is applicable also to the distributed file management system of the gestalt of operation shown by the 7th
 [0246] Moreover, although the Management Department of distributed file management system was explained as composition concentrated on one management server computer, you may make it form a management server computer in the distributed file management system of drawing 27 mentioned above for every computer site of every server computer of a predetermined group, and a predetermined group. If it does in this way, concentration of the load to the management server computer in a large-scale system can be prevented.

[0247] Moreover, you may make it choose at random the server computer which you may make it use a server computer in order based on a predetermined rule, and it not only chooses it as

order with only small the "load" of a server computer, but has the "load" below a predetermined threshold in the gestalt of operation mentioned above in selection of a partial file in case a partial file exists in two or more server **** on a plane.

[0248]

[Effect of the Invention] As mentioned above, since arrangement of a partial file was determined at each Management Department of a client computer to the server computer demanded according to the demand of creation of the distributed file to a server computer, reference, or updating or a management server computer based on the load information on each server computer according to the distributed file management equipment and distributed file management system of this invention, concentration of the load to a specific server computer could be avoided.

[0249] Moreover, since the external load information which notified load information to other server computers, and was notified from other server computers was held and arrangement of a partial file was determined based on the load information on other server computers, concentration of the load to a specific server computer could be avoided.

[0250] Moreover, since the partial file to which it is made to move was determined based on the access information, load information, and external load information for every partial file and a partial file was moved to other server computers, the imbalance of the capacity of concentration of the load to the storage of a specific server computer and the storage of each server computer could be avoided.

[0251] Moreover, since the partial file to copy was determined based on the access situation, load information, and external load information for every partial file and a partial file was copied to other server computers, concentration of the load to the storage of a specific server computer could be avoided.

[0252] Moreover, since the size of the partial file which divides and creates a distributed file according to the kind of data recorded on information and a distributed file from a client computer determines, the size of the partial file which constitutes a distributed file can change suitably, and it could prevent dividing and recording in the data which have relation content-wise and logically, for example, the data for one picture etc., to two or more partial files.

[0253] Moreover, since duplication of a resource was suppressed to the minimum by centralizing each Management Department which manages the partial file of a distributed file on one or more administrative server computers, the increase in cost could be suppressed.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

- [Drawing 1] It is the block diagram showing the distributed file management system of this invention.
- [Drawing 2] It is drawing showing the composition of the distributed file in this invention.
- [Drawing 3] It is the flow chart which shows the distributed file creation algorithm of the distributed file management section of this invention.
- [Drawing 4] It is drawing showing an example of the load information table in this invention.
- [Drawing 5] It is drawing showing an example of the distributed file control table of this invention.
- [Drawing 6] It is drawing showing an example of the partial file control table of this invention.
- [Drawing 7] It is the flow chart which shows reference/updating algorithm of the distributed file of the distributed file management section of this invention.
- [Drawing 8] It is the block diagram showing the distributed file management system of this invention.
- [Drawing 9] It is drawing showing an example of the external load information table in this invention.
- [Drawing 10] It is the flow chart which shows the distributed file creation algorithm of the distributed file management section in this invention.
- [Drawing 11] It is the block diagram showing the distributed file management system of this invention.
- [Drawing 12] It is drawing showing an example of the partial file access information table in this invention.
- [Drawing 13] It is drawing showing an example of the partial file control table in this invention.
- [Drawing 14] It is drawing showing an example of the partial file control table in this invention.
- [Drawing 15] It is the flow chart which shows the algorithm of the distributed file move section in this invention of operation.
- [Drawing 16] It is drawing showing an example of the partial file control table in this invention.
- [Drawing 17] It is drawing showing an example of the partial file control table in this invention.
- [Drawing 18] It is the flow chart which shows the algorithm of the distributed file move section in this invention of operation.
- [Drawing 19] It is drawing showing an example of the initial-entry table in this invention.
- [Drawing 20] It is the block diagram showing the distributed file management system of this invention.
- [Drawing 21] It is the flow chart which shows the algorithm of the distributed file copy section in this invention of operation.
- [Drawing 22] It is drawing showing an example of the partial file control table in this invention.
- [Drawing 23] It is drawing showing an example of the partial file control table in this invention.
- [Drawing 24] It is the flow chart which shows the algorithm of the distributed file copy section in this invention of operation.
- [Drawing 25] It is the flow chart which shows the algorithm of the distributed file copy section in this invention of operation.

- [Drawing 26] It is the block diagram showing the distributed file management system of this invention.
- [Drawing 27] It is the block diagram showing the distributed file management system of this invention.
- [Drawing 28] It is the flow chart which shows the distributed file creation algorithm of the distributed file management section of this invention.
- [Drawing 29] It is the flow chart which shows reference/updating algorithm of the distributed file of the distributed file management section of this invention.
- [Drawing 30] It is the block diagram showing conventional distributed file management equipment.

[Description of Notations]

- 101 Network
- 102-104, 802-804, 1102-1104, 2002-2004, 2602-2604, and 2702- 2704 and 2710 Computer site
- 105-107, 805-807, 1105-1107, 2005-2007, 2605-2607, 2705-2707 Server computer
- 108, 109, 110 Client computer group
- 108-1 - 108-n Client computer
- 111, 116, 121 Partial file management section
- 112, 117, 122, 1112, 1117, 1122, 2012, 2612, 2617, 2622, 2712 Distributed file management section
- 113, 118, 123, 2713 Network interface
- 114, 119, 124, 814, 819, 824, 2714 State Management Department
- 115, 120, 125 Storage
- 126a-128f, 202-1 - 202-n Partial file
- 131-133 Internal network
- 401 Load Information Table
- 402 Storage Load Information Table
- 403 Network Load Information Table
- 501 Distributed File Control Table
- 601, 1301, 1401-1403, 1601-1603, 1701-1703, 2201-2203, 2301-2303 Partial file control table
- 811, 812, 813 External state Management Department
- 901 External Load Information Table
- 1131, 1132, 1133 Distributed file move section
- 1201 Access Information Table
- 1901 Initial-Entry Table
- 2031, 2032, 2033 Distributed file copy section
- 2631, 2632, 2633 Partial file size determination section
- 2711 Management Server Computer

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2000-207370
(P2000-207370A)

(43) 公開日 平成12年7月28日 (2000.7.28)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード (参考)
G 0 6 F 15/177	6 7 4	G 0 6 F 15/177	6 7 4 A 5 B 0 4 5
12/00	5 4 5	12/00	5 4 5 B 5 B 0 8 2
15/16	6 2 0	15/16	6 2 0 B

審査請求 未請求 請求項の数31 O L (全 42 頁)

(21) 出願番号 特願平11-11513

(22) 出願日 平成11年1月20日 (1999.1.20)

(71) 出願人 000005821

松下電器産業株式会社
大阪府門真市大字門真1006番地

(72) 発明者 佐藤 正樹

大阪府門真市大字門真1006番地 松下電器
産業株式会社内

(72) 発明者 上杉 明夫

大阪府門真市大字門真1006番地 松下電器
産業株式会社内

(74) 代理人 100107526

弁理士 鈴木 直郁 (外1名)

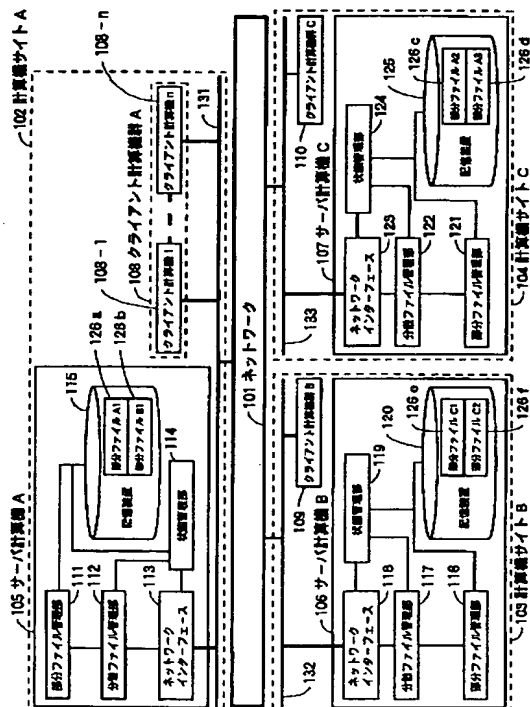
最終頁に続く

(54) 【発明の名称】 分散ファイル管理装置及び分散ファイル管理システム

(57) 【要約】

【課題】 ファイルの作成、参照、更新において、複数のサーバ計算機で適切な負荷分散を行うことができる分散ファイル管理システムを提供する。

【解決手段】 本発明の分散ファイル管理システムは、サーバ計算機A、B、Cと、クライアント計算機群108~110と、ネットワーク101とを備えている。サーバ計算機A105は、部分ファイルを記録する記憶装置115と、ネットワークインタフェース113と、部分ファイルの書き込みや読み出しを制御する部分ファイル管理部111と、負荷を監視し、負荷情報を保持する状態管理部114と、分散ファイル管理部112とによって構成されている。各サーバ計算機A、B、Cの負荷情報に基づいて部分ファイルの配置を決定するため、特定のサーバ計算機への負荷の集中を回避することができる。



【特許請求の範囲】

【請求項1】データを記憶する記憶手段を有する複数のサーバ計算機と1または複数のクライアント計算機とが接続されているネットワークに接続された分散ファイル管理装置であって、

前記複数のサーバ計算機の負荷情報を保持して管理する状態管理手段と、

前記クライアント計算機からの分散ファイルの処理要求に対応して、前記分散ファイルの部分ファイルを特定し、前記状態管理手段で管理されている前記負荷情報に基づいて、前記部分ファイルを処理するサーバ計算機を決定する分散ファイル管理手段と、

を備える、ことを特徴とする分散ファイル管理装置。

【請求項2】前記状態管理手段は、他の分散ファイル管理装置へ前記負荷情報を通知し、また、他の分散ファイル管理装置から通知されたサーバ計算機の負荷情報を外部負荷情報として保持する外部状態管理手段を備える、ことを特徴とする請求項1記載の分散ファイル管理装置。

【請求項3】前記外部状態管理手段は、マルチキャストによって前記他の分散ファイル管理装置へ前記負荷情報を通知する、ことを特徴とする請求項2記載の分散ファイル管理装置。

【請求項4】前記外部状態管理手段は、前記他の分散ファイル管理装置のうち、隣接する他の分散ファイル管理装置に前記負荷情報を通知する、ことを特徴とする請求項2または3記載の分散ファイル管理装置。

【請求項5】複数のサーバ計算機と、1または複数のクライアント計算機と、前記複数のサーバ計算機及び前記1または複数のクライアント計算機を接続するネットワークとを備えた分散ファイル管理システムにおいて、前記複数のサーバ計算機の各々は、

分散ファイルの一部または全部を構成する部分ファイルを記憶する記憶手段と、

負荷情報を保持して管理する状態管理手段と、

前記クライアント計算機からの分散ファイルの処理要求に対応して、前記分散ファイルの部分ファイルを特定し、前記状態管理手段で管理されている前記負荷情報に基づいて、前記部分ファイルを処理するサーバ計算機を決定する分散ファイル管理手段と、

を備える、

ことを特徴とする分散ファイル管理システム。

【請求項6】前記状態管理手段は、他のサーバ計算機へ前記負荷情報を通知し、また、他のサーバ計算機から通知された当該他のサーバ計算機の負荷情報を外部負荷情報として保持する外部状態管理手段を備える、ことを特徴とする請求項5記載の分散ファイル管理システム。

【請求項7】前記外部状態管理手段は、マルチキャストによって前記他のサーバ計算機へ前記負荷情報を通知する、ことを特徴とする請求項6記載の分散ファイル管理

システム。

【請求項8】前記複数のサーバ計算機は、1または複数のサーバ計算機群にグループ分けされており、

前記外部状態管理手段は、前記1または複数のサーバ計算機群のうち、所定のサーバ計算機群に属する他のサーバ計算機へ前記負荷情報を通知する、

ことを特徴とする請求項6または7記載の分散ファイル管理システム。

【請求項9】前記複数のサーバ計算機は、1または複数のサーバ計算機群にグループ分けされており、

前記外部状態管理手段は、前記1または複数のサーバ計算機群のうち、隣接するサーバ計算機群に属する他のサーバ計算機に前記負荷情報を通知する、

ことを特徴とする請求項6または7記載の分散ファイル管理システム。

【請求項10】前記分散ファイル管理手段は、前記部分ファイル毎のアクセス情報、前記負荷情報、及び前記外部負荷情報に基づいて移動する部分ファイルと移動先の他のサーバ計算機を決定し、該部分ファイルを該サーバ計算機へ移動する分散ファイル移動手段を備える、ことを特徴とする請求項2乃至4記載の分散ファイル管理装置または6乃至9記載の分散ファイル管理システム。

【請求項11】前記分散ファイル移動手段は、前記負荷情報に含まれる前記記憶手段の負荷が所定の値よりも大であることを検知し、前記外部負荷情報と前記アクセス情報に基づいて移動する部分ファイルと移動先の他のサーバ計算機を決定し、該部分ファイルを該他のサーバ計算機へ移動する、ことを特徴とする請求項10記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項12】前記分散ファイル移動手段は、前記負荷情報に含まれる前記記憶手段の残容量が所定の値よりも小であることを検知し、前記外部負荷情報と前記アクセス情報に基づいて移動する部分ファイルと移動先の他のサーバ計算機を決定し、該部分ファイルを該他のサーバ計算機へ移動する、ことを特徴とする請求項10記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項13】前記分散ファイル移動手段は、前記負荷情報に含まれる前記ネットワークの負荷が所定の値よりも大であることを検知し、前記外部負荷情報と前記アクセス情報に基づいて移動する部分ファイルと移動先の他のサーバ計算機を決定し、該部分ファイルを該他のサーバ計算機へ移動する、ことを特徴とする請求項10記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項14】前記分散ファイル移動手段は、前記負荷情報、前記外部負荷情報、前記アクセス情報、及び前記クライアント計算機と前記複数のサーバ計算機との間の接続情報に基づいて、前記部分ファイルを保持している記憶手段を有するサーバ計算機と処理要求を行ったクライアント計算機との間の通信コストを求め、該通信コスト

トよりも小の通信コストとなる他のサーバ計算機を決定し、該他のサーバ計算機へ前記部分ファイルを移動する、ことを特徴とする請求項10記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項15】前記分散ファイル移動手段は、前記部分ファイルの移動先となる前記他のサーバ計算機に対して、予め前記部分ファイルの移動が可能か否かを確認する、ことを特徴とする請求項10乃至14記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項16】前記分散ファイル移動手段は、前記部分ファイルを前記他のサーバ計算機へ移動した際に、前記他のサーバ計算機から前記サーバ計算機に他の部分ファイルを移動する、ことを特徴とする請求項10乃至15記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項17】前記分散ファイル移動手段は、前記部分ファイルを移動することができる他のサーバ計算機の候補をリストにし、該リストに基づいて、前記部分ファイルを移動する他のサーバ計算機を決定する、ことを特徴とする請求項10乃至16記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項18】前記分散ファイル移動手段は、前記部分ファイルの移動と共に、前記部分ファイルを作成したサーバ計算機に関する情報を前記他のサーバ計算機に送る、ことを特徴とする請求項10乃至17記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項19】前記分散ファイル管理手段は、前記部分ファイル毎のアクセス情報、前記負荷情報、及び前記外部負荷情報に基づいて、コピーする部分ファイルとコピー先の他のサーバ計算機を決定し、前記部分ファイルを前記他のサーバ計算機にコピーする分散ファイルコピー手段を備える、ことを特徴とする請求項2乃至4、6乃至18記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項20】前記分散ファイルコピー手段は、前記負荷情報に含まれる前記記憶手段の負荷が所定の値よりも大であることを検知し、前記外部負荷情報と前記アクセス情報に基づいてコピーする部分ファイルとコピー先の他のサーバ計算機を決定し、該部分ファイルを該他のサーバ計算機へコピーする、ことを特徴とする請求項19記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項21】前記分散ファイルコピー手段は、前記負荷情報に含まれる前記ネットワークの負荷が所定の値よりも大であることを検知し、前記外部負荷情報と前記アクセス情報に基づいてコピーする部分ファイルと移動先の他のサーバ計算機を決定し、該部分ファイルを該他のサーバ計算機へコピーする、ことを特徴とする請求項19記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項22】前記分散ファイルコピー手段は、前記負荷情報、前記外部負荷情報、前記アクセス情報、及び前記クライアント計算機と前記複数のサーバ計算機との間の接続情報に基づいて、前記部分ファイルを保持している記憶手段を有するサーバ計算機と処理要求を行ったクライアント計算機との間の通信コストを求め、該通信コストよりも小の通信コストとなる他のサーバ計算機を決定し、該他のサーバ計算機へ前記部分ファイルをコピーする、ことを特徴とする請求項19記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項23】前記分散ファイルコピー手段は、前記部分ファイルのコピー先となる前記他のサーバ計算機に対して、予め前記部分ファイルのコピーが可能か否かを確認する、ことを特徴とする請求項19乃至22記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項24】前記分散ファイルコピー手段は、前記部分ファイルを前記他のサーバ計算機へコピーした際に、前記他のサーバ計算機から前記サーバ計算機に他の部分ファイルをコピーする、ことを特徴とする請求項19乃至23記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項25】前記分散ファイルコピー手段は、前記部分ファイルをコピーすることができる他のサーバ計算機の候補をリストにし、該リストに基づいて、前記部分ファイルをコピーする他のサーバ計算機を決定する、ことを特徴とする請求項19乃至24記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項26】前記分散ファイルコピー手段は、前記部分ファイルをコピーする複数のコピー先の他のサーバ計算機を選択し、選択された前記複数の他のサーバ計算機へマルチキャストによって同時に前記部分ファイルをコピーする、ことを特徴とする請求項19乃至25記載の分散ファイル管理装置及び分散ファイル管理システム。

【請求項27】前記状態管理手段で管理されている前記負荷情報は、前記記憶手段の容量及び負荷、並びに前記ネットワークと前記複数のサーバ計算機との間の通信負荷を含む、ことを特徴とする請求項1乃至26記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項28】さらに、前記サーバ計算機は、前記部分ファイルを前記記憶手段に書き込み、また、前記部分ファイルを前記記憶手段から読み出す部分ファイル管理手段を備える、ことを特徴とする請求項1乃至27記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項29】前記分散ファイル管理手段は、前記クライアント計算機からの前記処理要求が分散ファイルの作成要求の場合には、該分散ファイルを複数の部分ファイルに分割し、分割した部分ファイルを保持するサーバ計算機を前記状態管理手段で管理されている前記負荷情報に基づいて決定し、

前記クライアント計算機からの前記処理要求が分散ファイルの参照要求または更新要求の場合には、前記参照要求または前記更新要求の処理の対象となる部分ファイルの存在を決め、前記処理要求を処理するサーバ計算機を前記状態管理手段で管理されている前記負荷情報に基づいて決定する、

ことを特徴とする請求項1乃至28記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項30】前記分散ファイル管理手段は、前記クライアント計算機からの情報に基づいて、分散ファイルの一部または全部を構成する前記部分ファイルのサイズを決定する部分ファイルサイズ決定手段を備える、ことを特徴とする請求項1乃至29記載の分散ファイル管理装置または分散ファイル管理システム。

【請求項31】前記分散ファイル管理手段は、分散ファイルに記録されているデータの種類に基づいて、前記分散ファイルの一部または全部を構成する前記部分ファイルのサイズを決定する部分ファイルサイズ決定手段を備える、ことを特徴とする請求項1乃至29記載の分散ファイル管理装置または分散ファイル管理システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、コンピュータネットワークシステムにおいて複数の端末にファイルを分散して管理する分散ファイル管理装置及び分散ファイル管理システムに関する。特に、複数のサーバ計算機及びクライアント計算機をネットワークで接続したサーバ・クライアント型のコンピュータネットワークシステムにおいて、複数のサーバ計算機でファイルを分散して管理する分散ファイル管理装置及び分散ファイル管理システムに関する。

【0002】

【従来の技術】従来から、複数のサーバ計算機及びクライアント計算機をネットワークで接続したサーバ・クライアント型のコンピュータネットワークシステム（以下、単に「ネットワークシステム」ともいう）に適用される分散ファイル管理技術として、例えば、特開平8-77054号広報に開示されている分散ファイルシステムなどがある。

【0003】図30は、特開平8-77054号広報に開示されている従来の分散ファイルシステムを示す。図30において、この従来の分散ファイルシステムは、複数のサーバ計算機3002、3003、3004、3005と、複数のクライアント計算機3006、3007、3008と、これら複数のサーバ計算機3002、3003、3004、3005及び複数のクライアント計算機3006、3007、3008を接続するネットワーク3001と、を備えている。

【0004】ここで、サーバ計算機3002には、分散ファイルAの部分ファイルA-1（3002-1）と分

散ファイルBの部分ファイルB-1（3002-3）が保持されている。また、サーバ計算機3003には、分散ファイルAの部分ファイルA-2（3003-1）と分散ファイルBの部分ファイルB-2（3003-2）が保持されている。また、サーバ計算機3004には、分散ファイルAの部分ファイルA-3（3004-1）と分散ファイルBの部分ファイルB-3（3004-2）が保持されている。

【0005】また、サーバ計算機3005は、各サーバ計算機3002、3003、3004に保持されているそれぞれの部分ファイルA-1～A-3、B-1～B-3を管理する分散ファイル管理部3005-2を備え、クライアント計算機3006、3007、3008からの部分ファイルの参照要求または更新要求（以下、単に「参照／更新要求」ともいう）に対してその振り分けを行なうための参照／更新要求振り分け情報3005-1を保持している。

【0006】一方、クライアント計算機3006は、分散ファイル作成要求に応じて分散ファイルの部分ファイルを作成する分散ファイル作成部3006-1と、分散ファイルに対する更新要求に応じて、サーバ計算機3005の参照／更新要求振り分け情報3005-1に基づいて該分散ファイルの部分ファイルの所在を決定する更新要求振り分け部3006-2と、分散ファイルに対する参照要求に応じて、サーバ計算機3005の参照／更新要求振り分け情報3005-1に基づいて該分散ファイルの部分ファイルの所在を決定する参照要求振り分け部3006-3と、を備えている。なお、他のクライアント計算機3007、3008も同様の構成になっている。

【0007】上述した従来の分散ファイルシステムによれば、例えば、クライアント計算機3006の利用者が分散ファイルの作成を要求した場合、クライアント計算機3006の分散ファイル作成部3006-1は、予め決められている振り分け条件に基づいて当該分散ファイルの部分ファイルを作成するサーバ計算機を決定し、該サーバ計算機に分散ファイルの部分ファイルを作成する。そして、この部分ファイルの作成と同時に、どのサーバ計算機に部分ファイルを作成したかを表す参照／更新要求振り分け情報3005-1を生成する。この参照／更新要求振り分け情報3005-1は、ネットワーク3001を介してサーバ計算機3005に送信され、サーバ計算機3005で保持される。

【0008】また、例えば、クライアント計算機3006の利用者が分散ファイルを参照または更新する参照／更新要求を行なった場合、まず、クライアント計算機3006は、サーバ計算機3005の分散ファイル管理部3005-2に対して、該当する分散ファイルのオープン要求を行なう。サーバ計算機3005の分散ファイル管理部3005-2は、クライアント計算機3006か

らの分散ファイルのオープン要求に対応して、クライアント計算機3006へネットワーク3001を介して当該分散ファイルに関する参照／更新要求振り分け情報3005-1を送信する。クライアント計算機3006の更新要求振り分け部3006-2または参照要求振り分け部3006-3は、サーバ計算機3005から受け取った参照／更新要求振り分け情報3005-1に基づいて、分散ファイルの部分ファイルを保持しているサーバ計算機に対し、参照／更新要求を送信する。

【0009】このように、従来の分散ファイルシステムにおいては、分散ファイルを複数の部分ファイルに分割し、分散ファイルに対する処理（作成、参照、更新）を部分ファイル単位の処理に分散することにより、1つの分散ファイルに対して複数の処理要求が集中した場合にも、1つのサーバ計算機に負荷を集中させずに、負荷の分散を行なうことができる。

【0010】

【発明が解決しようとする課題】しかしながら、図30に示したような従来の分散ファイルシステムにおいては、分散ファイルを構成する部分ファイルを作成する際に、予め決められている固定的な振り分け規則に基づいて、該部分ファイルを作成するサーバ計算機を決定しているため、サーバ計算機へのファイルの分散において現実的なサーバ計算機の負荷情報を考慮したものにはなっていない。このため、実際には、アクセスや処理が集中して負荷の高くなっているサーバ計算機に対して、さらに部分ファイルの作成要求が発生することがある。そのため、特定のサーバ計算機のみ負荷が大きくなる場合があり、複数のサーバ計算機で適切な負荷分散が行なわれないという問題があった。

【0011】また、従来の分散ファイルシステムにおいては、上記の振り分け規則に基づいて、一度固定的に部分ファイルの振り分けをサーバ計算機に行なってしまうと、その振り分け以後、すなわち、振り分けられたサーバ計算機での部分ファイルの作成以後には、作成された部分ファイルの移動やコピーを行なわないため、特定の部分ファイルへのアクセスが集中した場合、アクセスによる負荷の分散を行なうことができないという問題があった。

【0012】したがって、本発明の目的は、ファイルの作成、参照、更新において、複数のサーバ計算機で適切な負荷分散を行うことができる分散ファイル管理装置及び分散ファイル管理システムを提供することである。

【0013】

【課題を解決するための手段】上記課題を解決するために、本発明に係る第1の態様の分散ファイル管理装置は、データを記憶する記憶手段を有する複数のサーバ計算機と1または複数のクライアント計算機とが接続されているネットワークに接続された分散ファイル管理装置であって、複数のサーバ計算機の負荷情報を保持し

て管理する状態管理手段と、クライアント計算機からの分散ファイルの処理要求に対応して、分散ファイルの部分ファイルを特定し、状態管理手段で管理されている負荷情報に基づいて、部分ファイルを処理するサーバ計算機を決定する分散ファイル管理手段と、を備えることを特徴とする。

【0014】上述の本発明に係る分散ファイル管理装置において、状態管理手段は、他の分散ファイル管理装置へ負荷情報を通知し、また、他の分散ファイル管理装置から通知されたサーバ計算機の負荷情報を外部負荷情報として保持する外部状態管理手段を備えることもできる。

【0015】ここで、外部状態管理手段は、マルチキャストによって他の分散ファイル管理装置へ負荷情報を通知するようにしてもよい。また、外部状態管理手段は、他の分散ファイル管理装置のうち、隣接する他の分散ファイル管理装置に負荷情報を通知することもできる。

【0016】また、上記課題を解決するために、本発明に係る第1の態様の分散ファイル管理システムは、複数のサーバ計算機と、1または複数のクライアント計算機と、複数のサーバ計算機及び1または複数のクライアント計算機を接続するネットワークとを備えた分散ファイル管理システムにおいて、複数のサーバ計算機の各々は、分散ファイルの一部または全部を構成する部分ファイルを記憶する記憶手段と、負荷情報を保持して管理する状態管理手段と、クライアント計算機からの分散ファイルの処理要求に対応して、分散ファイルの部分ファイルを特定し、状態管理手段で管理されている負荷情報に基づいて、部分ファイルを処理するサーバ計算機を決定する分散ファイル管理手段と、を備えることを特徴とする。

【0017】上述の本発明に係る分散ファイル管理システムにおいて、状態管理手段は、他のサーバ計算機へ負荷情報を通知し、また、他のサーバ計算機から通知された当該他のサーバ計算機の負荷情報を外部負荷情報として保持する外部状態管理手段を備えるようにしてもよい。ここで、外部状態管理手段は、マルチキャストによって他のサーバ計算機へ負荷情報を通知することもできる。

【0018】また、上述の本発明に係る分散ファイル管理装置及び分散ファイル管理システムにおいて、複数のサーバ計算機は、1または複数のサーバ計算機群にグループ分けされており、外部状態管理手段は、1または複数のサーバ計算機群のうち、所定のサーバ計算機群に属する他のサーバ計算機へ負荷情報を通知するようにしてもよく、または、外部状態管理手段は、1または複数のサーバ計算機群のうち、隣接するサーバ計算機群に属する他のサーバ計算機に前記負荷情報を通知するようにしてもよい。

【0019】さらに、分散ファイル管理手段は、部分フ

ファイル毎のアクセス情報、負荷情報、及び外部負荷情報に基づいて移動する部分ファイルと移動先の他のサーバ計算機を決定し、該部分ファイルを該サーバ計算機へ移動する分散ファイル移動手段を備えることもできる。

【0020】ここで、分散ファイル移動手段は、負荷情報に含まれる記憶手段の負荷が所定の値よりも大であることを検知し、外部負荷情報とアクセス情報に基づいて移動する部分ファイルと移動先の他のサーバ計算機を決定し、該部分ファイルを該他のサーバ計算機へ移動するようにしてもよく、または、分散ファイル移動手段は、負荷情報に含まれる記憶手段の残容量が所定の値より小であることを検知し、外部負荷情報とアクセス情報に基づいて移動する部分ファイルと移動先の他のサーバ計算機を決定し、該部分ファイルを該他のサーバ計算機へ移動するようにしてもよい。また、分散ファイル移動手段は、負荷情報に含まれるネットワークの負荷が所定の値よりも大であることを検知し、外部負荷情報とアクセス情報に基づいて移動する部分ファイルと移動先の他のサーバ計算機を決定し、該部分ファイルを該他のサーバ計算機へ移動することもでき、または、分散ファイル移動手段は、負荷情報、外部負荷情報、アクセス情報、及びクライアント計算機と複数のサーバ計算機との間の接続情報に基づいて、部分ファイルを保持している記憶手段を有するサーバ計算機と処理要求を行ったクライアント計算機との間の通信コストを求め、該通信コストよりも小の通信コストとなる他のサーバ計算機を決定し、該他のサーバ計算機へ部分ファイルを移動するようにしてもよい。

【0021】さらに、分散ファイル移動手段は、部分ファイルの移動先となる他のサーバ計算機に対して、予め前記部分ファイルの移動が可能か否かを確認することもでき、また、分散ファイル移動手段は、部分ファイルを他のサーバ計算機へ移動した際に、他のサーバ計算機からサーバ計算機に他の部分ファイルを移動することもできる。さらに、分散ファイル移動手段は、部分ファイルを移動することができる他のサーバ計算機の候補をリストにし、該リストに基づいて、部分ファイルを移動する他のサーバ計算機を決定するようにしてもよい。また、分散ファイル移動手段は、部分ファイルの移動と共に、部分ファイルを作成したサーバ計算機に関する情報を他のサーバ計算機に送るようにすることもできる。

【0022】また、上述の分散ファイル管理手段は、部分ファイル毎のアクセス情報、負荷情報、及び外部負荷情報に基づいて、コピーする部分ファイルとコピー先の他のサーバ計算機を決定し、部分ファイルを他のサーバ計算機にコピーする分散ファイルコピー手段を備えることもできる。

【0023】このとき、分散ファイルコピー手段は、負荷情報に含まれる記憶手段の負荷が所定の値よりも大であることを検知し、外部負荷情報とアクセス情報に基づ

いてコピーする部分ファイルとコピー先の他のサーバ計算機を決定し、該部分ファイルを該他のサーバ計算機へコピーするようにしてもよく、分散ファイルコピー手段は、負荷情報に含まれるネットワークの負荷が所定の値よりも大であることを検知し、外部負荷情報とアクセス情報に基づいてコピーする部分ファイルと移動先の他のサーバ計算機を決定し、該部分ファイルを該他のサーバ計算機へコピーするようにしてもよい。または、分散ファイルコピー手段は、負荷情報、外部負荷情報、アクセス情報、及びクライアント計算機と複数のサーバ計算機との間の接続情報に基づいて、部分ファイルを保持している記憶手段を有するサーバ計算機と処理要求を行ったクライアント計算機との間の通信コストを求め、該通信コストよりも小の通信コストとなる他のサーバ計算機を決定し、該他のサーバ計算機へ部分ファイルをコピーするようにすることもできる。

【0024】ここで、分散ファイルコピー手段は、部分ファイルのコピー先となる他のサーバ計算機に対して、予め部分ファイルのコピーが可能か否かを確認するようにするとよい。また、分散ファイルコピー手段は、部分ファイルを他のサーバ計算機へコピーした際に、他のサーバ計算機からサーバ計算機に他の部分ファイルをコピーしてもよい。また、分散ファイルコピー手段は、部分ファイルをコピーすることができる他のサーバ計算機の候補をリストにし、該リストに基づいて、部分ファイルをコピーする他のサーバ計算機を決定するようにすることもできる。さらに、分散ファイルコピー手段は、部分ファイルをコピーする複数のコピー先の他のサーバ計算機を選択し、選択された複数の他のサーバ計算機へマルチキャストによって同時に部分ファイルをコピーすることもできる。

【0025】また、状態管理手段で管理されている負荷情報は、記憶手段の容量及び負荷、並びにネットワークと複数のサーバ計算機との間の通信負荷を含むことができる。

【0026】さらに、上述の本発明に係る分散ファイル管理装置及び分散ファイル管理システムにおいて、サーバ計算機は、部分ファイルを記憶手段に書き込み、また、部分ファイルを記憶手段から読み出す部分ファイル管理手段を備えることもできる。

【0027】また、分散ファイル管理手段は、クライアント計算機からの処理要求が分散ファイルの作成要求の場合には、該分散ファイルを複数の部分ファイルに分割し、分割した部分ファイルを保持するサーバ計算機を状態管理手段で管理されている負荷情報に基づいて決定し、クライアント計算機からの処理要求が分散ファイルの参照要求または更新要求の場合には、参照要求または更新要求の処理の対象となる部分ファイルの存在を決め、処理要求を処理するサーバ計算機を状態管理手段で管理されている負荷情報に基づいて決定する、ようにし

てもよい。

【0028】また、分散ファイル管理手段は、クライアント計算機からの情報に基づいて、分散ファイルの一部または全部を構成する部分ファイルのサイズを決定する部分ファイルサイズ決定手段を備えるようにすることができ、または、分散ファイル管理手段は、分散ファイルに記録されているデータの種別に基づいて、分散ファイルの一部または全部を構成する部分ファイルのサイズを決定する部分ファイルサイズ決定手段を備えるようにしてもよい。

【0029】上述の本発明に係る分散ファイル管理装置及び分散ファイル管理システムにおいては、分散ファイル管理手段が、サーバ計算機の負荷情報に基づいて、部分ファイルを配置するサーバ計算機を決定するため、特定のサーバ計算機への負荷の集中を回避することができる。

【0030】また、分散ファイル管理手段が、他のサーバ計算機の負荷情報に基づいて、部分ファイルを配置するサーバ計算機を決定するため、特定のサーバ計算機で負荷が集中することを回避できる。

【0031】また、部分ファイルを他のサーバ計算機に移動することによって、特定のサーバ計算機の記憶手段への負荷の集中や、記憶手段の容量の不均衡を回避することができる。また、部分ファイルを他のサーバ計算機にコピーすることによって、特定のサーバ計算機の記憶装置への負荷の集中を回避することができる。

【0032】また、分散ファイルを構成する部分ファイルのサイズを適宜変更することができるため、論理的、内容的に関連のあるデータ、例えば、画像1フレーム分のデータなどを複数の部分ファイルに分割してしまうことを避けられる。

【0033】

【発明の実施の形態】以下、本発明の分散ファイル管理装置及び分散ファイル管理システムの実施の形態について、図1から図29を用いて説明する。

【0034】（実施の形態1）図1は、本発明における分散ファイル管理システムの第1の実施の形態の一例を示す構成図である。図1において、この分散ファイル管理システムは、パーソナルコンピュータやワークステーションなどのサーバ計算機及びパーソナルコンピュータやワークステーションなどの複数のクライアント計算機から成るクライアント計算機群を備えた複数の計算機サイトA102、計算機サイトB103、及び計算機サイトC104と、計算機サイトA102、計算機サイトB103、及び計算機サイトC104を相互に接続するローカルエリアネットワークやワイドエリアネットワークなどのネットワーク101とを備えている。

【0035】ここで、計算機サイトA102は、パーソナルコンピュータやワークステーションなどの複数のサーバ計算機（図1においては、「サーバ計算機A10

5」のみ示す）と、パーソナルコンピュータやワークステーションなどのクライアント計算機1～n（108-1～108-n）から成るクライアント計算機群A108とを備えている。この計算機サイトA102は、複数のサーバ計算機（図1においては、「サーバ計算機A105」のみ示す）とクライアント計算機群A108とをイーサネットなどの内部ネットワーク131で接続しており、例えば、インターネットドメインになっている。

【0036】また、計算機サイトA102と同様に、計算機サイトB103は、複数のサーバ計算機（図1においては、「サーバ計算機B106」のみ示す）と、複数のクライアント計算機から成るクライアント計算機群B109とを備え、計算機サイトC104は、複数のサーバ計算機（図1においては、「サーバ計算機C107」のみ示す）と、複数のクライアント計算機から成るクライアント計算機群C110とを備えている。さらに、これらの計算機サイトB103及び計算機サイトC104は、計算機サイトA102と同様に、複数のサーバ計算機（図1においては、「サーバ計算機B106」及び「サーバ計算機C107」のみ示す）と、クライアント計算機群B109及びクライアント計算機群C110とを、それぞれ内部ネットワーク132及び内部ネットワーク133で接続しており、例えば、インターネットドメインになっている。

【0037】サーバ計算機A105は、分散ファイルの部分ファイルを記録するハードディスクなどの記憶装置115と、イーサネットなどの内部ネットワーク131へ接続するためのネットワークインタフェース113と、部分ファイルを記録している記憶装置115への書き込みや読み出しを制御する部分ファイル管理部111と、記憶装置115に対する負荷や記憶装置115の残り容量、及びネットワークインタフェース113に対する負荷を監視し、これらの負荷や容量に関する情報を保持する状態管理部114と、部分ファイル管理部111、状態管理部114、及びネットワークインタフェース113に接続された分散ファイル管理部112とによって構成されている。

【0038】この分散ファイル管理部112は、部分ファイルの書き込みや読み出しを部分ファイル管理部111に指示する。また、分散ファイル管理部112は、分散ファイルを作成する場合には、状態管理部114から得られる情報に基づいて、分散ファイルを複数の部分ファイルに分割し、各部分ファイルを配置（記録）するサーバ計算機を決定する。また、以前に作成された分散ファイルを参照または更新する場合には、該当する分散ファイルの部分ファイルが存在する（記録されている）サーバ計算機を決定する。

【0039】サーバ計算機B106及びサーバ計算機C107は、サーバ計算機A105と同様の構成になっている。すなわち、サーバ計算機B106は、記憶装置1

20と、ネットワークインタフェース118と、部分ファイル管理部116と、状態管理部119と、分散ファイル管理部117とによって構成されている。また、サーバ計算機C107は、記憶装置125と、ネットワークインタフェース123と、部分ファイル管理部121と、状態管理部124と、分散ファイル管理部122とによって構成されている。

【0040】図2は、分散ファイルの構成の一例を示す図である。図2において、分散ファイル201は、複数の部分ファイル202-1～202-nによって構成されている。

【0041】図1においては、各分散ファイルA、B、Cが作成された後の状態を示している。すなわち、サーバ計算機A105の記憶装置115には、分散ファイルAの部分ファイルA1(126a)と分散ファイルBの部分ファイルB1(126b)とが記録されている。また、サーバ計算機B106の記憶装置120には、分散ファイルCの部分ファイルC1(126e)と分散ファイルCの部分ファイルC2(126f)とが記録されている。また、サーバ計算機C107の記憶装置125には、分散ファイルAの部分ファイルA2(126c)と分散ファイルAの部分ファイルA3(126d)とが記録されている。

【0042】次に、以上のように構成された分散ファイル管理システムの動作について説明する。以下においては、クライアント計算機群A108のクライアント計算機1(108-1)からサーバ計算機A105に対して分散ファイルAの作成要求が発行され、図1に示したような部分ファイルA1～A3が作成される場合の分散処理を例にして説明する。ここで、図1に示した記憶装置115、120、125は、それぞれ複数の記憶部または記憶領域(以下、単に「記憶部」ともいう)を有するものとする。これらの複数の記憶部は、物理的に1つの記録媒体であってもよく、また、複数の記録媒体であってもよい。

【0043】図1において、まず、クライアント計算機1(108-1)からサーバ計算機A105に分散ファイルAの作成要求が発行される。この分散ファイルAの作成要求は、内部ネットワーク131及びサーバ計算機A105のネットワークインタフェース113を介して、サーバ計算機A105の分散ファイル管理部112によって受け取られる。

【0044】図3は、分散ファイルの作成要求を受け取った場合の分散ファイル管理部の動作アルゴリズムを示すフローチャートである。以下、図1及び図3を用いて、分散ファイル管理部112の詳細な動作を説明する。

【0045】分散ファイルAの作成要求を受け取った分散ファイル管理部112は、まず、状態管理部114の管理している負荷情報を獲得する(ステップ301)。

【0046】図4は、状態管理部114で管理する負荷情報テーブル401を示す図である。図4において、負荷情報テーブル401は、記憶装置負荷情報テーブル402及びネットワーク負荷情報テーブル403から成っている。

【0047】この記憶装置負荷情報テーブル402は、サーバ計算機A105に接続されている記憶装置115の複数の記憶部を識別するための「記憶装置識別子」と、各記憶部の負荷情報[%]を示す「負荷」と、各記憶部の残容量[Mbytes]を示す「残容量」の項目で構成されている。ここで、記憶装置115の各記憶部の「負荷」は、記憶装置115の各記憶部の最大転送レートのうち何%を使用しているかを示している。

【0048】また、ネットワーク負荷情報テーブル403は、ネットワークインタフェース113を介してネットワーク101上に送出するデータが、どの計算機サイト(送出先サイト)に向けて送出され、どの程度の帯域幅(使用通信帯域幅[Mbps])を使用しているか、また、受信しているデータがどの計算機サイト(送出元サイト)から送られて来たものであり、どの程度の帯域幅(使用通信帯域幅[Mbps])を使用して受信しているかを示している。この「送出元サイト」の項目がデータの送出元の計算機サイトを示し、「送出先サイト」の項目がデータの送出先の計算機サイトを示している。すなわち、Site Aは計算機サイトA105を、Site Bは計算機サイトB106を、Site Cは計算機サイトC107を示している。ここで、送信元サイト及び送信先サイトを総称してネットワークリンク(以下、単に「リンク」ともいう)という。また、「使用通信帯域幅」の項目が、送出元サイトと送出先サイトとの間で使用されている通信帯域幅[Mbps]を示している。

【0049】すなわち、分散ファイル管理部112が分散ファイルAの作成要求を受け取った場合、分散ファイル管理部112は、例えば図4に示すように、状態管理部114から獲得した記憶装置負荷情報テーブル402の情報に基づいて、記憶装置識別子DiskID1の記憶部の負荷が20[%]で、残容量が10[Mbytes]であるという情報を得る(ステップ301)。

【0050】次に、分散ファイル管理部112は、状態管理部114から得られる記憶装置負荷情報テーブル402の情報に基づいて、サーバ計算機A105に接続されている記憶装置115の中から、残容量が十分に残っており、且つ負荷が所定の閾値より低い記憶部を選択して、該記憶部に対して部分ファイルを順に割り当てていく。ここで、閾値としては、例えば、80[%]などを使用するとよい。但し、この閾値は、記憶装置115の構成などに応じて適宜決定することができる。また、部分ファイルのサイズは、固定長が望ましく、全てのサーバ計算機で同一のサイズにするとよい。このとき、全ての部分ファイルA1～A3を記憶装置115の記憶部に

割り当てられた場合にはステップ304の処理を行う。一方、全ての部分ファイルA1～A3を記憶装置115の記憶部に割り当てられなかった場合には、ステップ303の処理を行う（ステップ302）。

【0051】本実施の形態（図1）の場合では、全ての部分ファイルA1～A3を記憶装置115の記憶部に割り当てられなかった（ステップ302）ので、分散ファイルAの部分ファイルA1（126a）をサーバ計算機A105に割り当て、残りの部分ファイルA2、A3を他のサーバ計算機C107に割り当てている。

【0052】分散ファイル管理部112は、ステップ302で割り当てられなかった部分ファイルA2、A3を他のサーバ計算機に割り当てるために、他のサーバ計算機に対して内部ネットワーク131及びネットワーク101を介して、部分ファイルA2、A3の作成を行なえるかどうかの問い合わせを行なう。問い合わせを受けた他のサーバ計算機では、自己の状態管理部の負荷情報テーブル401を調べ、部分ファイルA2、A3の作成が可能かどうかを返答する（ステップ303）。この部分ファイルの作成の問い合わせや返答の信号のやり取りは、各サーバ計算機の分散ファイル管理部がネットワークインタフェース及びネットワークを介して行うことになる。

【0053】本実施の形態（図1）の場合では、分散ファイル管理部112が、ネットワークインタフェース113及びネットワーク101を介して、部分ファイルA2（126c）及び部分ファイルA3（126d）の作成を行なえるかどうかを計算機サイトC104のサーバ計算機C107の分散ファイル管理部122に問い合わせる。サーバ計算機C107の分散ファイル管理部122は、状態管理部124から得られる負荷情報テーブル401に基づいて、上述のステップ302と同様な判断を行ない、部分ファイルA2（126c）及び部分ファイルA3（126d）の作成を行えるかどうかの返答をサーバ計算機A105に行なう（ステップ303）。図1の場合、サーバ計算機C107の記憶装置125には、部分ファイルA2（126c）及び部分ファイルA3（126d）を作成することができる。

【0054】次に、サーバ計算機Aの分散ファイル管理部112は、サーバ計算機Aの記憶装置115に割り当てられた部分ファイルA1と、他のサーバ計算機Cの記憶装置125に割り当てられた部分ファイルA2、A3とを管理するための情報を登録する（ステップ304）。

【0055】図5は、分散ファイル管理テーブル501を示す図である。また、図6は、部分ファイル管理テーブル601を示す図である。図5において、分散ファイル管理テーブル501は、分散ファイルを識別するための「分散ファイル識別子」と、分散ファイルを構成する部分ファイルを識別するための「部分ファイル識別子リ

スト」の項目で構成されている。また、図6において、部分ファイル管理テーブル601は、部分ファイルを識別するための「部分ファイル識別子」と、部分ファイルの所在地を示す「所在地」の項目で構成されている。ここで、図6に示した「部分ファイル識別子」は、図5で示した「部分ファイル識別子リスト」を構成する「部分ファイル識別子」に対応している。

【0056】本実施の形態（図1）の場合では、例えば、分散ファイルAについて見ると、図5において、分散ファイルAが、部分ファイルA1（126a）、部分ファイルA2（126c）、部分ファイルA3（126d）から構成されることを表している。また、図6において、部分ファイルA1（126a）の所在地が、「file:///siteA/serverA/DiskID1/（計算機サイトA102のサーバ計算機A105の記憶装置識別子DiskID1）」であり、部分ファイルA2（126c）の所在地が、「file:///siteC/serverC/DiskID2/（計算機サイトC104のサーバ計算機C107の記憶装置識別子DiskID2）」であり、部分ファイルA3（126d）の所在地が、「file:///siteC/serverC/DiskID2/（計算機サイトC104のサーバC107の記憶装置DiskID2）」であることを表している。

【0057】次に、分散ファイル管理部112は、サーバ計算機Aの記憶装置115に部分ファイルA1を作成する場合、部分ファイル管理部111を介して、クライアント計算機1（108-1）からのデータを記憶装置115に書き込み、分散ファイルAの部分ファイルA1の作成を行なう。また、他のサーバ計算機C107に部分ファイルA2、A3を記録する場合、分散ファイル管理部112は、記録を行なうサーバ計算機C107の分散ファイル管理部122に部分ファイルA2、A3の記録を依頼し、それと同時に、分散ファイルAの作成要求を行なったクライアント計算機1（108-1）に指示をして、クライアント計算機1（108-1）から記録を行なうサーバ計算機C107に、直接データを送信するようにする。依頼を受けたサーバ計算機C107では、分散ファイル管理部122が部分ファイル管理テーブル601へ部分ファイルA2、A3の登録を行なう。このようにして、他のサーバ計算機C107上に、部分ファイルA2、3が作成される（ステップ305）。

【0058】以上のようにして、分散ファイルAの部分ファイルA1（126a）の作成は、クライアント計算機1（108-1）からのデータを、サーバ計算機A105の記憶装置115に書き込むことによって行なわれる。また、分散ファイルAの部分ファイルA2（126c）及び部分ファイルA3（126d）の作成は、クライアント計算機1（108-1）からのデータを、直接サーバ計算機C107に送り、サーバ計算機C107の

記憶装置125に書き込むことによって行なわれる。

【0059】また、上述のステップ303において、他の全てのサーバ計算機で、部分ファイルの作成が不可能な場合には、分散ファイル管理部112は、分散ファイルの作成要求を行なったクライアント計算機1(108-1)に対して、分散ファイルの作成に失敗したことを通知する(ステップ306)。

【0060】一方、ステップ302で、分散ファイルの全ての部分ファイルの作成を、自己のサーバ計算機の記憶装置にできる場合、分散ファイル管理部112は、記憶装置115に割り当てられた分散ファイルを管理するための情報を、図5に示した分散ファイル管理テーブル501と図6に示した部分ファイル管理テーブル601に登録する(ステップ307)。

【0061】次に、分散ファイル管理部112は、部分ファイル管理部111を介して、クライアント計算機1(108-1)からのデータを記憶装置115に書き込み、全ての部分ファイルの作成を行なう(ステップ308)。

【0062】以上のように、本発明の分散ファイル管理システムによれば、分散ファイルの作成を、各サーバ計算機の負荷を考慮して行うため、適切に負荷分散ができるようになる。

【0063】次に、クライアント計算機群A108内のクライアント計算機1(108-1)からサーバ計算機A105に対して分散ファイル参照/更新要求が発行された場合について説明する。

【0064】まず、クライアント計算機1(108-1)から発行された分散ファイルAに対する参照/更新要求は、サーバ計算機A105において、ネットワークインタフェース113を介して、分散ファイル管理部112によって受け取られる。

【0065】図7は、分散ファイルの参照/更新要求を受け取った場合の分散ファイル管理部の動作アルゴリズムを示すフローチャートである。以下、図7を用いて、分散ファイル管理部112の詳細な動作を説明する。また、以下においては、分散ファイルAに対する参照/更新要求の処理を例にした具体的な動作についても説明する。

【0066】まず、分散ファイル管理部112は、クライアント計算機1(108-1)からの分散ファイルの参照/更新要求に基づいて、分散ファイル管理テーブル501と部分ファイル管理テーブル601から、参照/更新する部分ファイルを特定し、その部分ファイルの所在地を求める(ステップ701)。

【0067】ここで、クライアント計算機1(108-1)からの分散ファイルAに対する参照/更新要求の場合、分散ファイル管理部112は、分散ファイル管理テーブル501に基づいて、分散ファイルAが、部分ファイルA1(126a)、部分ファイルA2(126

c)、部分ファイルA3(126d)によって構成されていることが解る。また、部分ファイル管理テーブル601に基づいて、部分ファイルA1(126a)の所在地が、「file:///siteA/serverA/DiskID1/(計算機サイトA102のサーバ計算機A105の記憶装置識別子DiskID1)」であり、部分ファイルA2(126c)の所在地が、「file:///siteC/serverC/DiskID2/(計算機サイトC104のサーバ計算機C107の記憶装置識別子DiskID2)」であり、部分ファイルA3(126d)の所在地が、「file:///siteC/serverC/DiskID2/(計算機サイトC104のサーバC107の記憶装置DiskID2)」であることが解る。

【0068】分散ファイル管理部112は、ステップ701で得られた部分ファイルの所在地から、参照/更新を行なう全ての部分ファイルが自己のサーバ計算機A105の記憶装置115に存在するかどうか、あるいは一部または全部の部分ファイルが他のサーバ計算機に存在するかどうかの判定を行なう(ステップ702)。

【0069】ここで、分散ファイルAの場合、部分ファイルA1(126a)は、記憶装置115に存在し、部分ファイルA2(126c)及び部分ファイルA3(126d)は、計算機サイトC104のサーバ計算機C107の記憶装置125に存在することがわかる。

【0070】次に、全ての部分ファイルが自己のサーバ計算機A105の記憶装置115に存在しない場合(ステップ702)、ステップ701で得られた部分ファイルの所在地に基づいて、参照/更新を行なう部分ファイルが記録されている他のサーバ計算機に部分ファイルの存在を確かめる(ステップ703)。

【0071】ここで、分散ファイルAの場合には、部分ファイルA2(126c)及び部分ファイルA3(126d)の存在を、計算機サイトC104のサーバ計算機C107の分散ファイル管理部122に確認する。

【0072】ステップ703で部分ファイルの存在が確認されたら、分散ファイル管理部112は、参照/更新を行なう部分ファイルが、自己のサーバ計算機A105の記憶装置115に存在する場合には、クライアント計算機1(108-1)からの参照/更新要求に基づいて、部分ファイル管理部111を介して記憶装置115に存在する部分ファイルの読み出し(参照)や部分ファイルへの書き込み(更新)を行なう。また、参照/更新を行なう部分ファイルが、他のサーバ計算機の記憶装置に存在する場合、分散ファイル管理部112は、参照/更新を行なう部分ファイルを保持するサーバ計算機に該部分ファイルの参照/更新を要求する。これと同時に、分散ファイル管理部112は、参照/更新要求を行なったクライアント計算機1(108-1)が、参照/更新を行なう部分ファイルを保持するサーバ計算機に参

照／更新要求を直接行なうように指示する（ステップ704）。

【0073】ここで、分散ファイルAの場合、部分ファイルA1（126a）が計算機サイトA102のサーバ計算機A105に、部分ファイルA2（126c）及び部分ファイルA3（126d）が計算機サイトC104のサーバ計算機C107に存在している。部分ファイルA1（126a）に対する参照／更新要求は、分散ファイル管理部112が、部分ファイル管理部111を介して、記憶装置115に対して参照／更新処理を行なう。一方、部分ファイルA2（126c）及び部分ファイルA3（126d）に対する参照／更新要求は、参照／更新要求を行ったクライアント計算機1（108-1）とサーバ計算機C107との間で、直接行われることになる。

【0074】また、ステップ703で部分ファイルの存在が確認されなかった場合、分散ファイル管理部112は、分散ファイルの参照／変更要求を行ったクライアント計算機1（108-1）に、分散ファイルの参照／更新が失敗したことを通知する（ステップ705）。

【0075】一方、ステップ702で、全ての部分ファイルが自己のサーバ計算機A105の記憶装置115に存在する場合、分散ファイル管理部112は、クライアント計算機1（108-1）からの参照／更新要求に基づいて、部分ファイル管理部111を介して記憶装置115に存在する部分ファイルの読み出し（参照）や部分ファイルへの書き込み（更新）を行なう（ステップ706）。

【0076】以上のように、上述した実施の形態によれば、クライアント計算機からサーバ計算機への要求が分散ファイルの作成の場合には、分散ファイルを複数の部分ファイルに分割し、サーバ計算機の負荷情報に基づいて各々の部分ファイルを作成するサーバ計算機を部分ファイル毎に決定して、分散ファイルの作成処理を行っている。また、クライアント計算機からの要求が分散ファイルの参照／更新の場合には、分散ファイルを構成する部分ファイルが存在するサーバ計算機を特定し、1または複数のサーバ計算機上に分散して配置されている部分ファイルをクライアント計算機から1つの分散ファイルとして扱うようにする。このようにして、クライアント計算機からサーバ計算機への分散ファイルの作成／参照／変更要求の際に、特定のサーバ計算機への負荷の集中をなくすることができる。

【0077】（実施の形態2）図8は、本発明における分散ファイル管理システムの第2の実施の形態の一例を示す構成図である。この図8においては、図1と同様の構成には同一の符号を付している。図8に示した分散ファイル管理システムは、パーソナルコンピュータやワークステーションなどのサーバ計算機及びパーソナルコンピュータやワークステーションなどの複数のクライアン

ト計算機から成るクライアント計算機群を備えた複数の計算機サイトA802、計算機サイトB803、及び計算機サイトC804と、計算機サイトA802、計算機サイトB803、及び計算機サイトC804を相互に接続するローカルエリアネットワークやワイドエリアネットワークなどのネットワーク101とを備えている。

【0078】ここで、計算機サイトA802は、パーソナルコンピュータやワークステーションなどの複数のサーバ計算機（図8においては、「サーバ計算機A805」のみ示す）と、パーソナルコンピュータやワークステーションなどのクライアント計算機1～n（108-1～108-n）から成るクライアント計算機群A108とを備えている。この計算機サイトA802は、複数のサーバ計算機（図8においては、「サーバ計算機A805」のみ示す）とクライアント計算機群A108とをイーサネットなどの内部ネットワーク131で接続しており、例えば、インターネットドメインになっている。

【0079】また、計算機サイトA802と同様に、計算機サイトB803は、複数のサーバ計算機（図8においては、「サーバ計算機B806」のみ示す）と、複数のクライアント計算機から成るクライアント計算機群B109とを備え、計算機サイトC804は、複数のサーバ計算機（図8においては、「サーバ計算機C807」のみ示す）と、複数のクライアント計算機から成るクライアント計算機群C110とを備えている。さらに、これらの計算機サイトB803及び計算機サイトC804は、計算機サイトA802と同様に、複数のサーバ計算機（図8においては、「サーバ計算機B806」及び「サーバ計算機C807」のみ示す）と、クライアント計算機群B109及びクライアント計算機群C110とを、それぞれ内部ネットワーク132及び内部ネットワーク133で接続しており、例えば、インターネットドメインになっている。

【0080】サーバ計算機A805は、分散ファイルの部分ファイルを記録するハードディスクなどの記憶装置115と、イーサネットなどの内部ネットワーク131へ接続するためのネットワークインタフェース113と、部分ファイルを記録している記憶装置115への書き込みや読み出しを制御する部分ファイル管理部111と、記憶装置115に対する負荷や記憶装置115の残り容量、及びネットワークインタフェース113に対する負荷を監視し、これらの負荷や容量に関する情報を保持する状態管理部814と、部分ファイル管理部111、状態管理部814、及びネットワークインタフェース113に接続された分散ファイル管理部112とによって構成されている。

【0081】この状態管理部814は、他のサーバ計算機へ負荷情報を通知し、また、他のサーバ計算機から通知された外部負荷情報を保持する外部状態管理部811を備えている。

【0082】サーバ計算機B806及びサーバ計算機C807は、サーバ計算機A805と同様の構成になっている。すなわち、サーバ計算機B806は、記憶装置120と、ネットワークインタフェース118と、部分ファイル管理部116と、外部状態管理部812を備えた状態管理部819と、分散ファイル管理部117とによって構成されている。また、サーバ計算機C807は、記憶装置125と、ネットワークインタフェース123と、部分ファイル管理部121と、外部状態管理部813を備えた状態管理部824と、分散ファイル管理部122とによって構成されている。

【0083】ここで、図8に示した分散ファイル管理システムと図1に示した分散ファイル管理システムとの相違点は、図8に示した状態管理部814、819、824が、他のサーバ計算機へ負荷情報を通知し、他のサーバ計算機から通知された外部負荷情報を保持する外部状態管理部811、812、813を備えている点である。

【0084】図9は、外部状態管理部811、812、813で管理されている外部負荷情報テーブル901の一例を示す。図9において、外部負荷情報テーブル901は、サーバ計算機の所在地を示す「サーバ計算機所在地」と、サーバ計算機所在地で示されるサーバ計算機の記憶装置の負荷情報を示す「記憶装置負荷情報」の項目で構成されている。また、「記憶装置負荷情報」は、記憶装置を識別するための「記憶装置識別子」と、記憶装置の負荷を示す「負荷」と、記憶装置の残容量を示す「残容量」の項目で構成されている。

【0085】外部状態管理部811、812、813は、外部負荷情報テーブル901からの外部負荷情報を他のサーバ計算機へ通知するために、以下のような動作を行なう。

【0086】まず、外部状態管理部811、812、813は、定期的または所定のタイミングで状態管理部814、819、824に対して記憶装置負荷情報テーブル402に示される情報を問い合わせるか、または、状態管理部814、819、824から記憶装置負荷情報テーブル402に示される情報の状態変化を通知してもらうことにより、状態管理部814、819、824で管理されている記憶装置負荷情報テーブル402の情報を得る。

【0087】次に、外部状態管理部811、812、813は、それぞれのネットワークインタフェース113、118、123を介して、各サーバ計算機に、記憶装置負荷情報を通知する。通知を受けたサーバ計算機では、それぞれネットワークインタフェース113、118、123を介して、外部状態管理部811、812、813が、記憶装置負荷情報を受け取り、それぞれの外部負荷情報テーブル901にこの情報を記録していく。

【0088】以上のように構成された分散ファイル管理

システムについて、クライアント計算機群A108内のクライアント計算機1(108-1)からサーバ計算機A805に対して分散ファイルAの作成要求が発行された場合を例として説明する。

【0089】まず、クライアント計算機1(108-1)から発行された分散ファイルAの作成要求は、内部ネットワーク131及びサーバ計算機A805のネットワークインタフェース113を介して、分散ファイル管理部112によって受け取られる。

【0090】図10は、分散ファイルの作成要求を受け取ったときの分散ファイル管理部の動作アルゴリズムを示すフローチャートである。以下、図10を用いて、分散ファイル管理部の詳細な動作を説明する。本実施の形態では、第1の実施の形態で説明した図3に示すステップ302とステップ303の処理を統合して、1つのステップ1002で処理することができる。

【0091】まず、分散ファイル管理部112は、状態管理部814の管理している負荷情報テーブル401と、外部状態管理部811の管理している外部負荷情報テーブル901から各情報を獲得する(ステップ1001)。

【0092】この状態管理部114では、図4に示したような負荷情報テーブル401を管理している。図4において、負荷情報テーブル401は、記憶装置負荷情報テーブル402と、ネットワーク負荷情報テーブル403とによって構成されている。

【0093】この記憶装置負荷情報テーブル402は、サーバ計算機A105に接続されている記憶装置115の複数の記憶部を識別するための「記憶装置識別子」と、各記憶部の負荷情報[%]を示す「負荷」と、各記憶部の残容量[Mbytes]を示す「残容量」の項目で構成されている。ここで、記憶装置115の各記憶部の「負荷」は、記憶装置115の各記憶部の最大転送レートのうち何%を使用しているかを示している。

【0094】また、ネットワーク負荷情報テーブル403は、ネットワークインタフェース113を介してネットワーク101上に送出するデータが、どの計算機サイト(送出先サイト)に向けて送出され、どの程度の帯域幅(使用通信帯域幅[Mbps])を使用しているか、また、受信しているデータがどの計算機サイト(送出元サイト)から送られて来たものであり、どの程度の帯域幅(使用通信帯域幅[Mbps])を使用して受信しているかを示している。この「送出元サイト」の項目がデータの送出元の計算機サイトを示し、「送出先サイト」の項目がデータの送出先の計算機サイトを示している。すなわち、Site Aは計算機サイトA105を、Site Bは計算機サイトB106を、Site Cは計算機サイトC107を示している。ここで、送信元サイト及び送信先サイトを総称してネットワークリンク(以下、単に「リンク」ともいう)という。また、「使用通信帯域

幅」の項目が、送出元サイトと送出先サイトとの間で使用されている通信帯域幅 [Mbps] を示している。

【0095】また、外部状態管理部811は、図9に示したような外部負荷情報テーブル901を管理している。

【0096】ここで、分散ファイルAを作成する場合、分散ファイル管理部112は、状態管理部814の負荷情報テーブル401から、「記憶装置識別子」がDiskID1で示される記憶装置115の記憶部の「負荷」が20 [%] で、「残容量」が10 [Mbytes] であるという情報を得ることができる。また、分散ファイル管理部112は、外部状態管理部814の外部負荷情報テーブル901から、計算機サイトB803のサーバ計算機B806の「記憶装置識別子」がDiskID1で示される記憶装置120の記憶部の「負荷」が49 [%] で、「残容量」が1000 [Mbytes] であり、また、計算機サイトC804のサーバ計算機C807の「記憶装置識別子」がDiskID1で示される記憶装置125の記憶部の「負荷」が30 [%] で、「残容量」が3000 [Mbytes] であるという情報を得ることができる。

【0097】次に、分散ファイル管理部112は、まず、状態管理部814から得られる記憶装置負荷情報テーブル402に基づいて、サーバ計算機A805に接続されている記憶装置115の各記憶部の中から、「残容量」が所定の容量以上で、且つ「負荷」が所定の閾値より低いという条件を満たす記憶部を選択し、分散ファイルを分割した部分ファイルを、当該条件を満たす各記憶部に順に割り当てていく。全ての部分ファイルを記憶装置115に割り当てられない場合には、分散ファイル管理部112は、外部状態管理部811から得られる外部負荷情報テーブル901の情報に基づいて、記憶部の「残容量」が所定の容量以上で、且つ「負荷」が所定の閾値より低い記憶部の存在する記憶装置を持つ他のサーバ計算機を選択する。そして、まだ割り当てられていない部分ファイルを、当該他のサーバ計算機の記憶装置の記憶部に順に割り当て、この割り当てを当該他のサーバ計算機に通知する。そして、全ての部分ファイルがサーバ計算機の各記憶装置の記憶部に割り当てて作成できたか否かを判断する (ステップ1002)。

【0098】ここで、分散ファイルAを作成する場合、分散ファイル管理部112は、状態管理部814の記憶装置負荷情報テーブル401に基づいて、部分ファイルA1 (126a) をサーバ計算機A805の記憶装置115の所定の記憶部に割り当てる。また、分散ファイル管理部112は、外部状態管理部811の外部負荷情報テーブル901の記憶装置負荷情報に基づいて、部分ファイルA2 (126c) 及び部分ファイルA3 (126d) をサーバ計算機C807の記憶装置125の各記憶部に割り当てる。

【0099】このように、サーバ計算機A805は、外

部状態管理部811から他のサーバ計算機の負荷や残容量の情報を得ることにより、他のサーバ計算機に部分ファイルの作成が可能かどうかの問い合わせをすることなく、他のサーバ計算機の負荷や残容量を考慮しながら部分ファイルを配置 (記憶) するサーバ計算機を決めることができる。

【0100】次に、分散ファイル管理部112は、分散ファイルの管理情報を、図5に示したような分散ファイル管理テーブル501と、図6に示したような部分ファイル管理テーブル601に登録する (ステップ1003)。図5において、分散ファイル管理テーブル501は、分散ファイルを識別するための「分散ファイル識別子」と、分散ファイルを構成する部分ファイルを識別するための「部分ファイル識別子リスト」の項目で構成されている。また、図6において、部分ファイル管理テーブル601は、部分ファイルを識別するための「部分ファイル識別子」と、部分ファイルの所在地を示す「所在地」の項目で構成されている。ここで、図6に示した「部分ファイル識別子」は、図5で示した「部分ファイル識別子リスト」を構成する「部分ファイル識別子」に対応している。

【0101】ここで、分散ファイルAの場合、図5において、分散ファイルAが、部分ファイルA1 (126a)、部分ファイルA2 (126c)、部分ファイルA3 (126d) から構成されることを表している。また、図6において、部分ファイルA1 (126a) の所在地が、「file:///siteA/serverA/DiskID1/(計算機サイトA102のサーバ計算機A105の記憶装置識別子DiskID1)」であり、部分ファイルA2 (126c) の所在地が、「file:///siteC/serverC/DiskID2/(計算機サイトC104のサーバ計算機C107の記憶装置識別子DiskID2)」であり、部分ファイルA3 (126d) の所在地が、「file:///siteC/serverC/DiskID2/(計算機サイトC104のサーバC107の記憶装置DiskID2)」であることを表している。

【0102】次に、分散ファイル管理部112は、記憶装置115の記憶部に部分ファイルを記録する場合には、部分ファイル管理部111を介して、クライアント計算機1 (108-1) からのデータを記憶装置115の所定の記憶部に書き込む。また、他のサーバ計算機の記憶装置に部分ファイルを記録する場合には、分散ファイル管理部112は、記録を行なうサーバ計算機の分散ファイル管理部に部分ファイルの記録を依頼する。これと同時に、分散ファイル管理部112は、分散ファイルの作成要求を行なったクライアント計算機1 (108-1) に指示して、記録を行なうサーバ計算機に直接データを送信するように指示する。分散ファイル管理部112から依頼を受けたサーバ計算機では、クライアント計

算機1(108-1)から部分ファイルのデータを受け取って、記憶装置の所定の記憶部に記録する。また、該サーバ計算機の分散ファイル管理部は、部分ファイル管理部の部分ファイル管理テーブル601へ部分ファイルの情報の登録を行なう。このようにして、他のサーバ計算機上に、部分ファイルが作成される(ステップ1004)。

【0103】分散ファイルAの場合、部分ファイルA1(126a)の作成は、クライアント計算機1(108-1)からのデータを記憶装置115の所定の記憶部に書き込むことによって行なわれる。部分ファイルA2(126c)及び部分ファイルA3(126d)の作成は、クライアント計算機1(108-1)から所定のデータを直接サーバ計算機C807に送り、サーバ計算機C807の記憶装置125の所定の記憶部にそれぞれ書き込むことによって行なわれる。

【0104】一方、ステップ1002で、どのサーバ計算機にも部分ファイルを作成できない場合には、分散ファイル管理部112は、分散ファイル作成の要求を行なったクライアント計算機1(108-1)に対して、分散ファイルの作成に失敗したことを通知する(ステップ1005)。

【0105】以上、分散ファイルの作成について説明したが、クライアント計算機からサーバ計算機に対して分散ファイルの参照/更新要求が発行された場合については、第1の実施の形態の場合(図7)と同様である。

【0106】以上のように、本発明の第2の実施の形態においては、状態管理部814、819、824が、他のサーバ計算機へ負荷情報を通知し、また、他のサーバ計算機から通知された外部負荷情報を保持する外部状態管理部811、812、813を備えることにより、分散ファイル管理部112、117、122は、他のサーバ計算機の負荷情報に基づいて、分散ファイルの部分ファイルを配置するサーバ計算機を決定することができ、特定のサーバ計算機への負荷の集中を回避することができる。

【0107】なお、外部状態管理部811、812、813から各サーバ計算機に記憶装置負荷情報を通知する際には、ユニキャストやマルチキャストを用いるとよい。特に、マルチキャストを用いた場合、全サーバ計算機に記憶装置負荷情報を一斉に通知することができ、通知のための通信量を減らすことができる。

【0108】また、通知するサーバ計算機をあらかじめ複数のグループに分けておき、各グループに属するサーバ計算機の各々にユニキャストで通知することもでき、また、各グループに対してマルチキャストで通知することもできる。このようにして、通知のための通信量を減らすことができる。

【0109】さらに、通知するサーバ計算機を隣接するサーバ計算機、すなわち、ネットワークで直接に接続さ

れているサーバ計算機に限定して、ユニキャストあるいはマルチキャストによって通知することもできる。これにより、通知のための通信量を減らすことができる。

【0110】(実施の形態3)図11は、本発明における分散ファイル管理システムの第3の実施の形態の一例を示す構成図である。この図11においては、図8と同様の構成には同一の符号を付している。図11に示した分散ファイル管理システムは、パーソナルコンピュータやワークステーションなどのサーバ計算機及びパーソナルコンピュータやワークステーションなどの複数のクライアント計算機から成るクライアント計算機群を備えた複数の計算機サイトA1102、計算機サイトB1103、及び計算機サイトC1104と、計算機サイトA1102、計算機サイトB1103、及び計算機サイトC1104を相互に接続するローカルエリアネットワークやワイドエリアネットワークなどのネットワーク101とを備えている。

【0111】ここで、計算機サイトA1102は、パーソナルコンピュータやワークステーションなどの複数のサーバ計算機(図11においては、「サーバ計算機A1105」のみ示す)と、パーソナルコンピュータやワークステーションなどのクライアント計算機1~n(108-1~108-n)から成るクライアント計算機群A108とを備えている。この計算機サイトA1102は、複数のサーバ計算機(図11においては、「サーバ計算機A1105」のみ示す)とクライアント計算機群A108とをイーサネットなどの内部ネットワーク131で接続しており、例えば、インターネットドメインになっている。

【0112】また、計算機サイトA1102と同様に、計算機サイトB1103は、複数のサーバ計算機(図11においては、「サーバ計算機B1106」のみ示す)と、複数のクライアント計算機から成るクライアント計算機群B109とを備え、計算機サイトC1104は、複数のサーバ計算機(図11においては、「サーバ計算機C1107」のみ示す)と、複数のクライアント計算機から成るクライアント計算機群C110とを備えている。さらに、これらの計算機サイトB1103及び計算機サイトC1104は、計算機サイトA1102と同様に、複数のサーバ計算機(図11においては、「サーバ計算機B1106」及び「サーバ計算機C1107」のみ示す)と、クライアント計算機群B109及びクライアント計算機群C110とを、それぞれ内部ネットワーク132及び内部ネットワーク133で接続しており、例えば、インターネットドメインになっている。

【0113】サーバ計算機A1105は、分散ファイルの部分ファイルを記録するハードディスクなどの記憶装置115と、イーサネットなどの内部ネットワーク131へ接続するためのネットワークインタフェース113と、部分ファイルを記録している記憶装置115への書

き込みや読み出しを制御する部分ファイル管理部111と、記憶装置115に対する負荷や記憶装置115の残り容量、及びネットワークインタフェース113に対する負荷を監視し、これらの負荷や容量に関する情報を保持する状態管理部814と、部分ファイル管理部111、状態管理部814、及びネットワークインタフェース113に接続された分散ファイル管理部1112とによって構成されている。

【0114】この状態管理部814は、他のサーバ計算機へ負荷情報を通知し、また、他のサーバ計算機から通知された外部負荷情報を保持する外部状態管理部811を備えている。

【0115】また、分散ファイル管理部1112は、部分ファイル毎のアクセス情報と負荷情報テーブル401及び外部負荷情報テーブル901の情報とに基づいて、移動させる部分ファイルを決定し、他のサーバ計算機へ部分ファイルを移動させる分散ファイル移動部1131を備えている。

【0116】サーバ計算機B1106及びサーバ計算機C1107は、サーバ計算機A1105と同様の構成になっている。すなわち、サーバ計算機B1106は、記憶装置120と、ネットワークインタフェース118と、部分ファイル管理部116と、外部状態管理部812を備えた状態管理部819と、分散ファイル移動部1132を備えた分散ファイル管理部1117とによって構成されている。また、サーバ計算機C1107は、記憶装置125と、ネットワークインタフェース123と、部分ファイル管理部121と、外部状態管理部813を備えた状態管理部824と、分散ファイル移動部1133を備えた分散ファイル管理部1122とによって構成されている。

【0117】ここで、図11に示した分散ファイル管理システムと図8に示した分散ファイル管理システムとの相違点は、図11に示した分散ファイル管理部1112、1117、1122が、部分ファイル毎のアクセス情報と負荷情報テーブル401及び外部負荷情報テーブル901の情報とから、移動させる部分ファイルを決定し、他のサーバ計算機へ部分ファイルを移動させる分散ファイル移動部1131、1132、1133を備えている点である。

【0118】図12は、状態管理部814で管理されている部分ファイル毎のアクセス情報テーブル1201の一例を示す。図12において、このアクセス情報テーブル1201は、部分ファイルを識別するための「部分ファイル識別子」と、「単位時間あたりのアクセス情報」の項目で構成されている。また、「単位時間あたりのアクセス情報」は、部分ファイルにアクセスしているクライアント計算機が存在するサイトの情報である「アクセス元サイト識別子」と、部分ファイルへのアクセス回数を示す「アクセス回数」の項目から構成されている。こ

のアクセス情報テーブル1201は、状態管理部814によって、単位時間毎に更新され続ける。

【0119】図13は、分散ファイル管理部1112で管理されている部分ファイル管理テーブル1301の一例を示す。図13において、部分ファイル管理テーブル1301は、部分ファイルを識別するための「部分ファイル識別子」と、部分ファイルの所在地を示す「所在地」と、部分ファイルが最初に作成された所在地を示す「オリジナル所在地」の項目から構成されている。図13において、部分ファイルが作成された段階では、「所在地」と「オリジナル所在地」の示す情報は同一であるが、部分ファイルが他のサーバ計算機に移動するのに応じて、「所在地」の情報は変化する。図13で示した部分ファイル管理テーブル1301は、図6で示した部分ファイル管理テーブル601に、「オリジナル所在地」の項目を加えたものになっている。

【0120】以上のように構成された分散ファイル管理システムにおいて、分散ファイルA、分散ファイルB、及び分散ファイルCが、上述で示したようにしてサーバ計算機A1105で作成された後、各分散ファイルの部分ファイルを移動する処理について詳細に説明する。

【0121】図14は、サーバ計算機A1105によって作成された分散ファイルA、分散ファイルB、及び分散ファイルCの部分ファイル管理テーブル1301の内容の一例を示している。図14において、図14(A)は、サーバ計算機A1105の部分ファイル管理テーブル1401を示し、(B)はサーバ計算機B1106の部分ファイル管理テーブル1402を示し、(C)はサーバ計算機C1107の部分ファイル管理テーブル1403を示している。図14(A)の部分ファイル管理テーブル1401には、部分ファイル識別子が、A1、A2、A3、B1、C1、C2で示される各部分ファイルの所在地と、オリジナル所在地が示されている。部分ファイル管理テーブル1402には、部分ファイル識別子が、C1、C2で示される各部分ファイルの所在地と、オリジナル所在地が示されている。部分ファイル管理テーブル1403には、部分ファイル識別子が、A2、A3で示される各部分ファイルの所在地と、オリジナル所在地が示されている。ここで、図14においては、各部分ファイルの移動前の状態を表している。このため、全ての部分ファイルにおいて、その所在地とオリジナル所在地が一致している。

【0122】図14に示した状態での部分ファイルの移動の際の、サーバ計算機A1105の分散ファイル移動部1131の動作アルゴリズムについて説明する。

【0123】図15は、分散ファイル移動部1131の動作アルゴリズムを示す。まず、分散ファイル移動部1131は、状態管理部814が管理している負荷情報テーブル401(図4)の情報を、一定時間の間隔で監視する(ステップ1501)。

【0124】分散ファイル移動部1131は、ある記憶装置の「負荷」が予め設定されている所定の閾値（例えば、80%などの値で、この値は、システムの構成などによって任意に決定する）を越えたことを検出すると

（ステップ1501）、この検出された記憶装置に含まれている部分ファイルの「部分ファイル識別子」を、部分ファイル管理テーブル1301を参照して探す。探し出した「部分ファイル識別子」の「単位時間当たりのアクセス情報」を、アクセス情報テーブル1201から得る。ここで得られた「単位時間当たりのアクセス情報」の「アクセス回数」を各「部分ファイル識別子」毎に比較し、最も大きな「アクセス回数」になっている「部分ファイル識別子」を選択する（ステップ1502）。すなわち、移動元部分ファイルを選択する。例えば、ここで、移動元部分ファイルとして部分ファイルA1（126a）が選択されたとする。

【0125】次に、分散ファイル移動部1131は、外部負荷情報テーブル901（図9）に基づいて、「残容量」が十分で且つ「負荷」が所定の値より低い記憶装置を持つサーバ計算機を選択する。そして、分散ファイル移動部1131は、選択したサーバ計算機に対して部分ファイルが移動できるかどうかを確認し、移動可能なサーバ計算機を決定する（ステップ1502）。すなわち、移動先サーバ計算機を選択する。例えば、ここで、サーバ計算機C1107の記憶装置識別子DiskID2で示される記憶装置125の記憶部が選択されたとする。

【0126】次に、この移動元部分ファイルの選択と移動先サーバ計算機の選択によって得られた情報に基づいて、部分ファイルの移動を行なう（ステップ1503）。

【0127】上述の例では、ステップ1502で、移動元部分ファイルとして部分ファイルA1（126a）、移動先サーバ計算機としてサーバ計算機C1107が選択されているので、移動元のサーバ計算機A1105の分散ファイル移動部1131は部分ファイル管理部111を介して、記憶装置115から部分ファイルA1（126a）を読み出す。この読み出された部分ファイルA1（126a）は、部分ファイルA1（126a）の「オリジナル所在地」（図13）に関する情報と共に、ネットワークインタフェース113及び内部ネットワーク131を介してネットワーク101へ送出される。

【0128】一方、移動先のサーバ計算機C1107において、分散ファイル移動部1133は、移動元のサーバ計算機A1105より送出された部分ファイルA1（126a）とその「オリジナル所在地」の情報を、ネットワーク101から内部ネットワーク133及びネットワークインタフェース123を介して受信する。部分ファイル管理部121は、この受信した部分ファイルA1（126a）を記憶装置125に書き込む。また、部

分ファイルA1（126a）の「オリジナル所在地」を、部分ファイル管理部121の部分ファイル管理テーブル（図14（C））に登録する。

【0129】その後、移動先のサーバ計算機C1107は、移動元のサーバ計算機A1105と「オリジナル所在地」（図13）に示されているサーバ計算機（この例の場合では、サーバ計算機A1105、すなわち、移動元とオリジナルは同じサーバ計算機A1105）に対して、部分ファイルA1（126a）の移動が完了したことを通知する。移動元のサーバ計算機A1105と「オリジナル所在地」に示されるサーバ計算機では、部分ファイル管理テーブルに登録されている部分ファイルA1（126a）の情報を書き換える。

【0130】図16は、上述の例のように部分ファイルA1（126a）が移動した後の、図14に示した部分ファイル管理テーブル1401、1402、1403の状態を示す。図16において、図16（A）はサーバ計算機A1105の部分ファイル管理テーブル1601を示し、（B）はサーバ計算機B1106の部分ファイル管理テーブル1602を示し、（C）はサーバ計算機C1107の部分ファイル管理テーブル1603を示している。すなわち、図16（A）～（C）の各部分ファイル管理テーブル1601、1602、1603は、それぞれ図14（A）～（C）の各部分ファイル管理テーブル1401、1402、1403に対応している。ここで、部分ファイル管理テーブル1601には、「部分ファイル識別子」がA1、A2、A3、B1、C1、C2で示される各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。部分ファイル管理テーブル1602には、「部分ファイル識別子」がC1、C2で示される各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。部分ファイル管理テーブル1603には、「部分ファイル識別子」がA1、A2、A3で示される各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。ここで、図16に示した部分ファイル管理テーブル1601、1602、1603の状態と図14に示した部分ファイル管理テーブル1401、1402、1403の状態の差異は、部分ファイルA1（126a）を、サーバ計算機A1105からサーバ計算機C1107へ移動させたことによるものである。すなわち、図16と図14の相違点は、図14

（A）の部分ファイル管理テーブル1401において、部分ファイルA1（126a）の「所在地」が「file:///siteA/serverA/DiskID1/」であると登録されている情報が、図16（A）の部分ファイル管理テーブル1601においては、部分ファイルA1（126a）の「所在地」が「file:///siteC/serverC/DiskID2/」と登録されている点と、図16（C）の部分ファイル管理テーブル1603において部分ファイルA1の項目が追加

されている点である。

【0131】図17は、さらに、この図16に示した状態から、上述した処理と同様にして、サーバ計算機B1106の分散ファイル移動部1132が、部分ファイルC1(126e)を、サーバ計算機C1107に移動させ、サーバ計算機C1107の分散ファイル移動部1133が、部分ファイルA1(126a)を、サーバ計算機B1106に移動させた場合の部分ファイル管理テーブルを示す。

【0132】図17において、図17(A)はサーバ計算機A1105の部分ファイル管理テーブル1701を示し、(B)はサーバ計算機B1106の部分ファイル管理テーブル1702を示し、(C)はサーバ計算機C1107の部分ファイル管理テーブル1703を示している。部分ファイル管理テーブル1701には、「部分ファイル識別子」がA1、A2、A3、B1、C1、C2で示される各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。部分ファイル管理テーブル1702には、「部分ファイル識別子」がC1、C2、A1で示される各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。また、部分ファイル管理テーブル1703には、「部分ファイル識別子」がA2、A3、C1で示される各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。

【0133】ここで、図17に示した部分ファイル管理テーブル1701、1702、1703の状態と図16に示した部分ファイル管理テーブル1601、1602、1603の状態の差異は、部分ファイルA1(126a)を、サーバ計算機C1107からサーバ計算機B1106へ移動させたことと、部分ファイルC1(126e)をサーバ計算機B1106からサーバ計算機C1107へ移動させたによるものである。

【0134】すなわち、部分ファイルA1(126a)の移動に対応して、部分ファイル管理テーブル1702には、部分ファイルA1(126a)の項目が追加されている。また、部分ファイル管理テーブル1703においては、部分ファイルA1(126a)の項目(図16参照)が削除されている。さらに、サーバ計算機B1106が部分ファイルA1(126a)の「オリジナル所在地」を参照して、サーバ計算機A1105に移動を知らせ、この通知によって、「オリジナル所在地」で示されるサーバ計算機A1105においては、部分ファイル管理テーブル1701に登録されている部分ファイルA1(126a)の「所在地」を「file:///siteB/serverB/DiskID2/」に変更している。

【0135】また、部分ファイルC1(126e)の移動に対応して、部分ファイル管理テーブル1702に登録されている部分ファイルC1(126e)の「所在地」が、「file:///siteB/serverB

/DiskID3/」(図16(B))から、「file:///siteC/serverC/DiskID3/」に変更されている。また、部分ファイル管理テーブル1703においては、部分ファイルC1(126e)の項目が追加されている。

【0136】次に、図17に示す状態で、クライアント計算機1(108-1)が、分散ファイルCを参照する際に、その参照内容が部分ファイルC1(126e)に含まれる場合の動作を説明する。

【0137】(1)クライアント計算機1(108-1)は、分散ファイルCを作成したサーバ計算機A1105に対して、分散ファイルCの参照を要求する。サーバ計算機A1105では、分散ファイル管理テーブル1701を参照して、分散ファイルCを構成する部分ファイルC1、C2のうちどの部分ファイルを参照しているのかを調べる。ここでは、部分ファイルC1(126e)とする。部分ファイルC1(126e)の「所在地」は、「file:///siteB/serverB/DiskID3/」なので、サーバ計算機A1105は、サーバ計算機B1106に対して、部分ファイルC1(126e)が存在するかどうかの確認を行なう。

【0138】(2)サーバ計算機B1106は、分散ファイル管理テーブル1702を調べて、部分ファイルC1(126e)の「所在地」を調べる。部分ファイルC1(126e)の「所在地」は、「file:///siteC/serverC/DiskID3/」なので、サーバ計算機B1106は、サーバ計算機C1107に対して、部分ファイルC1(126e)が存在するかどうかの確認を行なう。

【0139】(3)サーバ計算機C1107は、分散ファイル管理テーブル1703を調べて、部分ファイルC1(126e)の「所在地」を調べる。部分ファイルC1(126e)の「所在地」は、「file:///siteC/serverC/DiskID3/」なので、部分ファイルC1(126e)は、サーバ計算機C1107に存在することが解る。

【0140】(4)サーバ計算機C1107は、サーバ計算機B1106に、部分ファイルC1(126e)は、「file:///siteC/serverC/DiskID3/」に存在することを通知する。

【0141】(5)この通知を受けて、サーバ計算機B1106は、サーバ計算機A1105に、部分ファイルC1(126e)が「file:///siteC/serverC/DiskID3/」に存在することを通知する。

【0142】(6)サーバ計算機A1105は、サーバ計算機C1107に部分ファイルC1(126e)の参照を要求する。この要求と同時に、参照を要求したクライアント計算機1(108-1)に対して、サーバ計算機C1107に直接、部分ファイルC1(126e)の

参照要求を行なうように指示する。また、サーバ計算機A1105では、部分ファイルC1(126e)の「所在地」を、「file:///siteC/serverC/DiskID3/」へと書き換える。

【0143】以上のように、本実施の形態では、分散ファイル管理部1112、1117、1122が、部分ファイル毎のアクセス情報テーブル1201、負荷情報テーブル401及び外部負荷情報テーブル901の各情報に基づいて、移動させる部分ファイルを決定する。また、他のサーバ計算機へ部分ファイルを移動する分散ファイル移動部1131、1132、1133を備えることにより、部分ファイルを他のサーバ計算機に移動することによって、特定のサーバ計算機の記憶装置への負荷の集中を回避できる。

【0144】なお、上述した第3の実施の形態では、分散ファイル移動部1131の動作アルゴリズム(図15)のステップ1501において、記憶装置の各記憶部の「負荷」が所定の値を越えたことを検知する代わりに、記憶装置の各記憶部の「残容量」が所定の値、例えば、10[Mbytes]などの値(ただし、この値は、装置やシステムの構成に応じて決定される)を下回ったことを検知するようにしてもよい。これによって、記憶装置の各記憶部の容量の不均衡を回避できる。

【0145】また、上述のステップ1501において、記憶装置の各記憶部の「負荷」が所定の値を越えたことを検知する代わりに、ネットワーク101の「負荷情報」から「使用通信帯域幅」が所定の値、例えば、使用可能通信帯域幅の80[%]の値(ただし、この値は、装置やシステムの構成に応じて決定される)を越えたリンクを検知するようにすることもできる。また、ステップ1502において、アクセス情報テーブル1201からネットワーク101の負荷を高めている部分ファイルと、ネットワークの負荷を高めている計算機サイトを選択することによって、ネットワークの負荷の集中を回避することができる。例えば、計算機サイトA1102

(送出元サイト)と計算機サイトB1103(送出先サイト)のリンクの「使用通信帯域幅」(図4)が所定の値を越えた時、サーバ計算機A1105に存在し、ネットワーク負荷を高める原因となっている部分ファイルを、サーバ計算機B1103に移動する。これにより、計算機サイトA1102と計算機サイトB1103間の「使用通信帯域幅」を減少させることができる。

【0146】また、上述の第3の実施の形態では、ステップ1502において、移動先のサーバ計算機の部分ファイルを移動可能かを確認し、ステップ1503において、部分ファイルの移動を行なっているが、事前にステップ1502で部分ファイルの移動可能かを確認することなく部分ファイルを移動することによって、ステップ1502の確認処理を省略することができる。このとき、部分ファイルの移動先のサーバ計算機側で、部分フ

ァイルの移動を受け入れられない場合には、移動先のサーバ計算機が、さらに部分ファイルを移動するための移動先を探し、この部分ファイルを移動するようにすればよい。

【0147】また、上述のステップ1503において、部分ファイルを移動元のサーバ計算機から、移動先のサーバ計算機へ移動しているが、その移動処理に加えて、移動先のサーバ計算機内の部分ファイルの中から移動元のサーバ計算機へ移動可能な他の部分ファイルを選択し、当該他の部分ファイルを移動元のサーバ計算機に移動するようにしてもよい。これによって、部分ファイルが1つのサーバ計算機に集中することを防ぐことができ、よりファイルアクセスに対する負荷を軽減することができる。

【0148】また、上述のステップ1502において、部分ファイルの移動先のサーバ計算機を選択を行なう際に、あらかじめサーバ計算機リストを設定し、リスト中のサーバ計算機の中から、記憶装置の各記憶部の残容量が十分あり、負荷が所定の値より低い記憶装置を持つサーバ計算機を選択するようにするとよい。これによって、部分ファイルの移動先のサーバ計算機を選択に費やされる時間を短縮することができる。

【0149】(実施の形態4)図18は、図11に示した分散ファイル管理システムの分散ファイル移動部1131、1132、1133の他の動作アルゴリズムを示すフローチャートである。

【0150】図18において、まず、分散ファイル移動部1131、1132、1133は、各部分ファイルへの通信コストを所定の間隔で監視する(ステップ1801)。ここで、通信コストとしては、例えば、部分ファイルを参照しているクライアント計算機と、その部分ファイルを保持するサーバ計算機との間の通信時間とすることができる。図11において、例えば、クライアント計算機1(108-1)と部分ファイルA2(126c)の通信コストは、部分ファイルA2(126c)を参照しているクライアント計算機1(108-1)と、部分ファイルA2(126c)を保持するサーバ計算機C1107との間の通信時間とする。

【0151】ここで、分散ファイル移動部1131、1132、1133は、部分ファイルへの通信コストが所定の値、例えば、1秒など(ただし、この値は、装置やシステムの構成に応じて決定される)を越えたことを検知する(ステップ1801)と、通信コストが所定の値を越えた部分ファイルを移動元の部分ファイルとして選択する(ステップ1802)。また、この通信コストが所定の値を越えた部分ファイルに対して、複数のクライアント計算機がアクセスしている場合、各々のアクセスに対する通信コストを求め、これらを加算して合計通信コストを求める。

【0152】移動先のサーバ計算機を選択する際には、

外部負荷情報テーブル901に基づいて、記憶装置の各記憶部の「残容量」が十分にあり、それらに対する「負荷」が所定の値より低い記憶装置を持つサーバ計算機を選択する。そして、選択したサーバ計算機に対して、上述の合計通信コストを送信し、部分ファイルを移動した結果、通信コストがどのように変化するかを順に問い合わせ、最小の通信コストになるサーバ計算機を選択する（ステップ1802）。または、分散ファイル移動部1131、1132、1133が、サイト間の接続情報を持ち、その接続情報から通信コストを予想して、最小の通信コストになるサーバ計算機を選択するようにしてもよい（ステップ1802）。

【0153】図19は、接続情報テーブルの一例を示す図である。図19において、接続情報テーブル1901は、部分ファイルを送出する「送出元サイト」と、部分ファイルが送出される「送出先サイト」と、送出元サイトから送出先サイトまでの通信コストを示す「通信時間」の項目を有する。この接続情報テーブル1901から、部分ファイルを保持するサーバ計算機がどのサイトに属し、部分ファイルを参照するクライアント計算機がどのサイトに属すかにより、サーバ計算機とクライアント計算機間の「通信時間」を得ることができる。この「通信時間」と上述した合計通信コストに基づいて、部分ファイルの移動後に「通信時間」が最小となるサーバ計算機を、部分ファイルの移動先として選択する（ステップ1802）。ここで、通信コストが所定の値を越えた部分ファイルに複数のクライアント計算機がアクセスしている場合には、部分ファイルの移動後の「通信時間」が最小の通信コストになるサーバ計算機を選択するようにするとよい。

【0154】例えば、クライアント計算機1（108-1）と部分ファイルA2（126c）の通信コスト（以下、「コストA2」ともいう）が所定の値を越えたとき、部分ファイルA2（126c）をサーバ計算機A1105に移動した場合に、通信コストがどのようになるかをサーバ計算機A1105に問い合わせるか、または、接続情報テーブル1901から通信コストを求め、その結果が、コストA2を下回っていれば、サーバ計算機A1105を移動先サーバ計算機の候補にする。この処理を他のサーバ計算機に対しても行い、通信コストが最小になるサーバ計算機を探す（ステップ1802）。

【0155】最後に、この移動元部分ファイルの選択と移動先サーバ計算機の選択（ステップ1802）によって得られた情報に基づいて、部分ファイルの移動を行なう（ステップ1803）。

【0156】以上のように、第4の実施の形態では、分散ファイル移動部1131、1132、1133が、負荷情報テーブル401、外部負荷情報テーブル901及びアクセス情報テーブル1201の各情報と、接続情報

テーブル1901から得られるサーバ計算機とクライアント計算機間の接続情報とに基づいて、処理中の部分ファイルが存在するサーバ計算機と処理の要求元のクライアント計算機との間の通信コストを求め、通信コストが所定の値を超えた場合に、通信コストの小くなる他のサーバ計算機へ部分ファイルを移動するようにしたので、上述した第3の実施の形態で得られる効果に加え、クライアント計算機から部分ファイルへのアクセス時間の平均値を短縮することができる。

【0157】なお、上述の第4の実施の形態では、通信コストとして、通信時間を例にあげているが、通信時間の「遅延」や「ゆらぎ（変動幅）」などにもすることができる。

【0158】また、上述の第4の実施の形態では、複数のクライアント計算機が同一の部分ファイルにアクセスしている場合、合計通信コストを最小にするように部分ファイルを移動しているが、平均通信コストを最小にするように部分ファイルを移動するようにしてもよい。

【0159】（実施の形態5）図20は、本発明における分散ファイル管理システムの第5の実施の形態の一例を示す構成図である。この図20においては、図8と同様の構成には同一の符号を付している。図20に示した分散ファイル管理システムは、パーソナルコンピュータやワークステーションなどのサーバ計算機及びパーソナルコンピュータやワークステーションなどの複数のクライアント計算機から成るクライアント計算機群を備えた複数の計算機サイトA2002、計算機サイトB2003、及び計算機サイトC2004と、計算機サイトA2002、計算機サイトB2003、及び計算機サイトC2004を相互に接続するローカルエリアネットワークやワイドエリアネットワークなどのネットワーク101とを備えている。

【0160】ここで、計算機サイトA2002は、パーソナルコンピュータやワークステーションなどの複数のサーバ計算機（図20においては、「サーバ計算機A2005」のみ示す）と、パーソナルコンピュータやワークステーションなどのクライアント計算機1～n（108-1～108-n）から成るクライアント計算機群A108とを備えている。この計算機サイトA2002は、複数のサーバ計算機（図20においては、「サーバ計算機A2005」のみ示す）とクライアント計算機群A108とをイーサネットなどの内部ネットワーク131で接続しており、例えば、インターネットドメインになっている。

【0161】また、計算機サイトA2002と同様に、計算機サイトB2003は、複数のサーバ計算機（図20においては、「サーバ計算機B2006」のみ示す）と、複数のクライアント計算機から成るクライアント計算機群B109とを備え、計算機サイトC2004は、複数のサーバ計算機（図20においては、「サーバ計算

機C2007」のみ示す)と、複数のクライアント計算機から成るクライアント計算機群C110とを備えている。さらに、これらの計算機サイトB2003及び計算機サイトC2004は、計算機サイトA2002と同様に、複数のサーバ計算機(図20においては、「サーバ計算機B2006」及び「サーバ計算機C2007」のみ示す)と、クライアント計算機群B109及びクライアント計算機群C110とを、それぞれ内部ネットワーク132及び内部ネットワーク133で接続しており、例えば、インターネットドメインになっている。

【0162】サーバ計算機A2005は、分散ファイルの部分ファイルを記録するハードディスクなどの記憶装置115と、イーサネットなどの内部ネットワーク131へ接続するためのネットワークインタフェース113と、部分ファイルを記録している記憶装置115への書き込みや読み出しを制御する部分ファイル管理部111と、記憶装置115に対する負荷や記憶装置115の残り容量、及びネットワークインタフェース113に対する負荷を監視し、これらの負荷や容量に関する情報を保持する状態管理部814と、部分ファイル管理部111、状態管理部814、及びネットワークインタフェース113に接続された分散ファイル管理部2012とによって構成されている。

【0163】この状態管理部814は、他のサーバ計算機へ負荷情報を通知し、また、他のサーバ計算機から通知された外部負荷情報を保持する外部状態管理部811を備えている。

【0164】また、分散ファイル管理部2012は、アクセス情報テーブル1201(図12)、負荷情報テーブル401(図4)及び外部負荷情報テーブル901

(図9)から得られる部分ファイル毎の各情報に基づいて、コピーする部分ファイルを決定し、他のサーバ計算機へ当該部分ファイルをコピーする分散ファイルコピー部2001を備えている。

【0165】サーバ計算機B2006及びサーバ計算機C2007は、サーバ計算機A2005と同様の構成になっている。すなわち、サーバ計算機B2006は、記憶装置120と、ネットワークインタフェース118と、部分ファイル管理部116と、外部状態管理部812を備えた状態管理部819と、分散ファイルコピー部2032を備えた分散ファイル管理部2017とによって構成されている。また、サーバ計算機C2007は、記憶装置125と、ネットワークインタフェース123と、部分ファイル管理部121と、外部状態管理部813を備えた状態管理部824と、分散ファイルコピー部2033を備えた分散ファイル管理部2022とによって構成されている。

【0166】ここで、図20に示した分散ファイル管理システムと図8に示した分散ファイル管理システムとの相違点は、図20に示した分散ファイル管理部201

2、2017、2022が、アクセス情報テーブル1201、負荷情報テーブル401及び外部負荷情報テーブル901から得られる部分ファイル毎の各情報に基づいて、コピーする部分ファイルを決定し、他のサーバ計算機へ当該部分ファイルをコピーする分散ファイルコピー部2031、2032、2033を備えている点である。

【0167】以上のように構成された分散ファイル管理システムの動作について、分散ファイルA、分散ファイルB及び分散ファイルCが、サーバ計算機A2005によって図20に示すように作成された後に、部分ファイルのコピーを行なう場合を例にとりて詳細に説明する。

【0168】図21は、サーバ計算機A2005の分散ファイルコピー部2031の動作アルゴリズムを示すフローチャートである。図21において、まず、分散ファイルコピー部2031は、状態管理部814が管理している負荷情報テーブル401(図4)を、所定の時間間隔で監視する(ステップ2101)。

【0169】分散ファイルコピー部2031は、記憶装置115の任意の記憶部の「負荷」が所定の値、例えば、80[%]などの値を越えたことを検出する(ステップ2101)と、検出された記憶装置115の記憶部に含まれている部分ファイルを、部分ファイル管理テーブル1301(図13)を参照して探し出す。そして探し出した部分ファイルのアクセス情報を、アクセス情報テーブル1201から獲得する。得られたアクセス情報の「アクセス回数」を比較し、最も大きな「アクセス回数」になっている部分ファイルを、コピー元部分ファイルとして選択する。ここで、例えば、部分ファイルA1(126a)が選択されたとする。次に、外部負荷情報テーブル901(図9)に基づいて、記憶装置の記憶部の「残容量」が十分あり、「負荷」が所定の値より低い記憶装置を持つサーバ計算機を選択する。そして、選択したサーバ計算機に対して、部分ファイルがコピーできるかどうかを確認し、コピー可能なサーバ計算機をコピー先サーバ計算機として選択する(ステップ2102)。

【0170】ここで、例えば、サーバ計算機C2007の記憶装置125の記憶部(記憶装置識別子:DiskID2)が選択されたとする。

【0171】このコピー元部分ファイルの選択とコピー先サーバ計算機を選択によって得られた情報に基づいて、部分ファイルのコピーを行ない(ステップ2103)、再びステップ2101の監視処理を続行する。

【0172】ここで、上述の例の場合、ステップ2102で、コピー元部分ファイルとして部分ファイルA1(126a)、コピー先サーバ計算機としてサーバ計算機C2007が選択されたので、コピー元のサーバ計算機A2005の分散ファイルコピー部2031は、部分ファイル管理部111を介して、記憶装置115から部

分ファイルA1 (126a)を読み出し、この部分ファイルA1 (126a)をネットワークインタフェース113を介してネットワーク101へ送出する。また、同時に部分ファイルA1 (126a)の「オリジナル所在地」(図13)に関する情報も送出する。

【0173】一方、コピー先のサーバ計算機C2007では、分散ファイルコピー部2033が、コピー元のサーバ計算機A2005より送出された部分ファイルA1 (126a)を、ネットワークインタフェース123を介して受信する。そして、部分ファイル管理部121を介して、記憶装置125の所定の記憶部に書き込む。また、部分ファイルA1 (126a)の「オリジナル所在地」も受信して、部分ファイル管理テーブル1403に登録する。その後、コピー先のサーバ計算機C2007は、コピー元のサーバ計算機A2005と「オリジナル所在地」に示されているサーバ計算機(この例の場合には、「オリジナル所在地」もサーバ計算機A2005の記憶装置115の記憶部である)に対して、部分ファイルA1 (126a)のコピーが完了したことを通知する。コピー元のサーバ計算機と「オリジナル所在地」に示されるサーバ計算機(両方ともサーバ計算機A2005)では部分ファイル管理テーブル1401の部分ファイルA1 (126a)の情報を書き換える。

【0174】図22は、コピー処理後のサーバ計算機の部分ファイル管理テーブルを示す図である。上述の部分ファイルA1 (126a)のコピーの結果、図14に示した部分ファイル管理テーブル1401、1402、1403は、図22に示す部分ファイル管理テーブル2201、2202、2203の状態に変化する。すなわち、図22(A)は、サーバ計算機A2005の部分ファイル管理テーブル2201を示し、(B)は、サーバ計算機B2006の部分ファイル管理テーブル2202を示し、(C)は、サーバ計算機C2007の部分ファイル管理テーブル2203を示している。また、部分ファイル管理テーブル2201には、「部分ファイル識別子」がA1、A2、A3、B1、C1及びC2で示される各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。また、部分ファイル管理テーブル2202には、「部分ファイル識別子」がC1及びC2で示される各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。また、部分ファイル管理テーブル2203には、「部分ファイル識別子」がA1、A2及びA3で示される各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。図14に示した部分ファイル管理テーブルの状態と、図22に示した部分ファイル管理テーブルの状態の差異は、部分ファイルA1 (126a)を、サーバ計算機A2005からサーバ計算機C2007へコピーしたことによる差異である。すなわち、図14(A)の部分ファイル管理テーブル1401において、部分ファイルA1 (126a)

の「所在地」が、「file:///siteA/serverA/DiskID1/」であるのに対して、図22(A)の部分ファイル管理テーブル2201においては、部分ファイルA1 (126a)の「所在地」が「file:///siteA/serverA/DiskID1/」及び「file:///siteC/serverC/DiskID2/ (コピーの所在地)」となっている点が相違する。さらに、図22(C)の部分ファイル管理テーブル2103においては、部分ファイルA1 (126a)の項目が追加されている。

【0175】また、図22の状態から、サーバ計算機B2006の分散ファイルコピー部2032が、部分ファイルC1 (126e)を、サーバ計算機C2007にコピーし、サーバ計算機C2007の分散ファイルコピー部2033が、部分ファイルA1 (126a)を、サーバ計算機B2006にコピーすることもできる。

【0176】図23は、図22の状態から、さらに部分ファイルをコピーした状態の部分ファイル管理テーブルを示す図である。図23(A)は、サーバ計算機A2005の部分ファイル管理テーブル2301を示し、

(B)は、サーバ計算機B2006の部分ファイル管理テーブル2302を示し、(C)は、サーバ計算機C2007の部分ファイル管理テーブル2303を示している。図23(A)の部分ファイル管理テーブル2301には、「部分ファイル識別子」がA1、A2、A3、B1、C1及びC2の各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。図23(B)の部分ファイル管理テーブル2302には、「部分ファイル識別子」がC1、C2及びA1の各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。また、図23(C)の部分ファイル管理テーブル2303には、「部分ファイル識別子」がA1、A2、A3及びC1の各部分ファイルの「所在地」と、「オリジナル所在地」が示されている。図22の部分ファイル管理テーブルの状態と図23の部分ファイル管理テーブルの状態の差異は、部分ファイルA1 (126a)を、サーバ計算機C2007からサーバ計算機B2006へコピーしたことと、部分ファイルC1 (126e)をサーバ計算機B2006からサーバ計算機C2007へコピーしたことによるものである。

【0177】すなわち、部分ファイルA1 (126a)のコピーに対応して、部分ファイル管理テーブル2302において、部分ファイルA1 (126a)の項目が追加されている。また、サーバ計算機B2006が部分ファイルA1 (126a)の「オリジナル所在地」を参照して、サーバ計算機A2005にコピーを知らせることで、「オリジナル所在地」のサーバ計算機A2005が、部分ファイル管理テーブル2301において、部分ファイルA1 (126a)の「所在地」に、「file:///siteB/serverB/DiskID2

／」を追加している。また、部分ファイルC1(126e)のコピーに対応して、部分ファイル管理テーブル2303において、部分ファイルC1(126e)の項目が追加されている。さらに、部分ファイル管理テーブル2302において、部分ファイルC1(126e)の「所在地」が、「file:///siteB/serverB/DiskID3/」および「file:///siteC/serverC/DiskID3/ (コピーの所在地)」となっている。

【0178】図23に示す状態で、クライアント計算機1(108-1)が、分散ファイルCを参照する際に、その参照内容が部分ファイルC1(126e)に含まれる場合の動作を説明する。

【0179】(1)クライアント計算機1(108-1)は、分散ファイルCを作成したサーバ計算機A2005に対して、分散ファイルCの参照を要求する。サーバ計算機A2005では、分散ファイル管理テーブル2301を参照して、分散ファイルCを構成する部分ファイルC1、C2のうちどの部分ファイルを参照しているのかを調べ、部分ファイルC1(126e)の参照であることを認識する。部分ファイルC1(126e)の「所在地」は、分散ファイル管理テーブル2301では「file:///siteB/serverB/DiskID3/」なので、サーバ計算機A2005は、サーバ計算機B2006に対して、部分ファイルC1(126e)が存在するかどうかの確認を行なう。

【0180】(2)サーバ計算機B2006は、分散ファイル管理テーブル2302を調べて、部分ファイルC1(126e)の所在地を調べる。部分ファイルC1(126e)の「所在地」は、分散ファイル管理テーブル2302では「file:///siteB/serverB/DiskID3/」及び「file:///siteC/serverC/DiskID3/」なので、サーバ計算機B2006は、負荷情報テーブル401と外部負荷情報テーブル901からサーバ計算機B2006とサーバ計算機C2007のうち負荷の低いサーバ計算機を選択する。ここで、サーバ計算機C2007を選択した場合、サーバ計算機B2006は、サーバ計算機C2007に対して、部分ファイルC1(126e)が存在するかどうかの確認を行なう。

【0181】(3)サーバ計算機C2007は、分散ファイル管理テーブル2303を調べて、部分ファイルC1(126e)の「所在地」を確認する。部分ファイルC1(126e)の「所在地」は、分散ファイル管理テーブル2303では「file:///siteC/serverC/DiskID3/」なので、部分ファイルC1(126e)は、サーバ計算機C2007に存在することが解る。

【0182】(4)サーバ計算機C2007は、サーバ計算機B2006に、部分ファイルC1(126e)

が、「file:///siteC/serverC/DiskID3/」に存在することを通知する。

【0183】(5)サーバ計算機B2006は、サーバ計算機A2005に、部分ファイルC1(126e)が、「file:///siteC/serverC/DiskID3/」に存在することを通知する。

【0184】(6)サーバ計算機A2005は、サーバ計算機C2007に部分ファイルC1(126e)の参照を要求する。この要求と同時に、参照を要求したクライアント計算機1(108-1)に対して、クライアント計算機1(108-1)がサーバ計算機C2007に直接に、部分ファイルC1(126e)の参照要求を行なうように指示する。

【0185】以上のように、本実施の形態では、分散ファイル管理部112、117、122が、アクセス情報テーブル1201、負荷情報テーブル401及び外部負荷情報テーブル901から得られる部分ファイル毎の各情報に基づいて、コピーする部分ファイルを決定し、分散ファイルコピー部2031、2032、2033が、部分ファイルを他のサーバ計算機にコピーすることによって、特定のサーバ計算機の記憶装置への負荷の集中を回避することができる。

【0186】なお、上述した第5の実施の形態では、分散ファイルコピー部2031の動作アルゴリズム(図21)のステップ2101において、記憶装置の各記憶部の「負荷」が所定の値を越えたことを検知するのに代えて、ネットワーク負荷情報テーブル403の「使用通信帯域幅」が所定の値を越えたリンクを検知することにしてもよい。また、ステップ2102において、アクセス情報テーブル1201からネットワーク負荷を高めている部分ファイルと、ネットワーク負荷を高めている計算機サイトを選択するようにしてもよい。これによって、ネットワーク負荷の集中を回避できる。例えば、計算機サイトA2002(送出元サイト)と計算機サイトB2003(送出先サイト)のリンクの「使用通信帯域幅」が所定の値を越えた時、サーバ計算機A2005に存在し、ネットワーク負荷を高める原因となっている部分ファイルをサーバ計算機B2003にコピーする。これにより、計算機サイトA2002と計算機サイトB2003間の「使用通信帯域幅」が減少する。

【0187】図24は、サーバ計算機A2005の分散ファイルコピー部2031の他の動作アルゴリズムを示すフローチャートである。図24において、まず、分散ファイルコピー部2031は、部分ファイルへの通信コストを所定の間隔で監視する(ステップ2401)。

【0188】ここで、通信コストには、例えば、部分ファイルを参照しているクライアント計算機と、その部分ファイルを保持するサーバ計算機との間の通信時間とすることができる。例えば、クライアント計算機1(108-1)と部分ファイルA2(126c)の通信コスト

の場合は、部分ファイルA2(126c)を参照しているクライアント計算機1(108-1)と、部分ファイルA2(126c)を保持するサーバ計算機C2007の間の通信時間とすればよい。

【0189】次に、分散ファイルコピー部2031は、部分ファイルへの通信コストが所定の値を越えたことを検知した場合(ステップ2401)、この通信コストが所定の値を越えた部分ファイルを、コピー元部分ファイルとして選択する。通信コストが所定の値を越えた部分ファイルに、複数のクライアント計算機がアクセスしている場合には、各アクセス毎に各々の通信コストを求め、これらを加算して合計通信コストを求める。一方、コピー先のサーバ計算機を選択する際には、外部負荷情報テーブル901に基づいて、記憶装置の記憶部の「残容量」が十分あり、「負荷」が低い記憶装置を持つサーバ計算機を選択し、選択されたサーバ計算機に対して合計通信コストを送信し、部分ファイルをコピーした結果、通信コストがどのように変化するかを順に問い合わせる。そして、最小の通信コストになるサーバ計算機をコピー先サーバ計算機として選択する。または、分散ファイルコピー部2031、2032、2033が、サイト間(サーバ計算機とクライアント計算機との間)の接続情報として、例えば、図19に示す接続情報テーブル1901を持ち、その情報から通信コスト(通信時間)を予想して、最小の通信コストになるサーバ計算機を選択するようにしてもよい。図19の接続情報テーブル1901の情報と、サーバ計算機がどのサイトに属しクライアント計算機がどのサイトに属すかの情報とにより、サーバ計算機とクライアント計算機の間「通信時間」を得ることができる。そして、その「通信時間」が最小となるサーバ計算機を、部分ファイルのコピー先として選択するとよい。通信コストが所定の値を越えた部分ファイルに、複数のクライアント計算機がアクセスしている場合には、最小の合計通信コストになるサーバ計算機をコピー先サーバ計算機として選択する(ステップ2402)。

【0190】例えば、クライアント計算機1(108-1)と部分ファイルA2の通信コスト(コストA2)が所定の値を越えた時には、部分ファイルA2をサーバ計算機A105にコピーした場合に、通信コストがどのようになるかをサーバ計算機A105に問い合わせる。あるいは、接続情報1801からコストを求める。その結果が、コストA2を下回っていれば、サーバ計算機A105をコピー先の候補にする。この処理を繰り返して、通信コストが最小になるサーバ計算機を探す。このコピー元部分ファイルの選択とコピー先サーバ計算機を選択によって得られた情報に基づいて、部分ファイルのコピーを行ない(ステップ2403)、再びステップ2401の監視処理を続行する。

【0191】このように、図21のステップ2101と

ステップ2102を図24のステップ2401とステップ2402に変更することで、クライアント計算機から部分ファイルへのアクセス時間の平均値を短縮することができる。また、上記では通信コストとして、「通信時間」を例にあげているが、通信時間の「遅延」や「ゆらぎ(変動幅)」などでもよい。

【0192】また、図21のステップ2102及び図24のステップ2402において、コピー先のサーバ計算機に部分ファイルのコピーが可能かを確認し、ステップ2103及びステップ2403において、部分ファイルのコピーを行なっているが、ステップ2102及びステップ2402でサーバ計算機に部分ファイルのコピーが可能かを確認することなく、部分ファイルをコピーすることによって、ステップ2102及びステップ2402の確認処理を省略することができる。このとき、ステップ2103及びステップ2403において、コピー先のサーバ計算機側で、部分ファイルのコピーを受け入れられない場合には、コピー先のサーバ計算機は、さらに部分ファイルをコピーするためのコピー先を探して部分ファイルをコピーするか、あるいは、コピー用に送信されてきた部分ファイルを破棄して、コピー元のサーバ計算機にコピー用の部分ファイルを破棄したことを通知するようにしてもよい。

【0193】また、上述の第5の実施の形態では、図21のステップ2103及び図24のステップ2403において、部分ファイルをコピー元のサーバ計算機から、コピー先のサーバ計算機へ単にコピーしているが、そのコピー処理に加えて、コピー先のサーバ計算機内の部分ファイルの中からコピー元のサーバ計算機へ移動させても良い部分ファイルを選択し、その部分ファイルを移動元のサーバ計算機に移動するようにしてもよい。これによって、1つのサーバ計算機に部分ファイルが集中せず、特定のサーバ計算機の記憶装置への負荷の集中を回避することができる。

【0194】また、図21のステップ2102において、コピー先のサーバ計算機を選択を行なう際に、あらかじめサーバ計算機リストを設定しておき、このサーバ計算機リスト中のサーバ計算機の中から、記憶装置の記憶部の残容量が十分あり、負荷が低い記憶装置を持つサーバ計算機を選択するようにすることもできる。これによって、サーバ選択のための時間を短縮することができるようになる。

【0195】(実施の形態6)図25は、図20に示した分散ファイル管理システムにおけるサーバ計算機A2005の分散ファイルコピー部2031の他の動作アルゴリズムを示すフローチャートである。本実施の形態においては、上述した第5の実施の形態と同様の構成となっており、分散ファイルコピー部の動作を除いては同様の動作を行う。以下、分散ファイルコピー部2031の動作アルゴリズムについて説明する。

【0196】図25において、まず、分散ファイルコピー部2031は、状態管理部814が管理している負荷情報テーブル401(図4)を、所定の時間間隔で監視する(ステップ2501)。

【0197】分散ファイルコピー部2031は、記憶装置115の任意の記憶部の「負荷」が所定の値、例えば、80[%]などの値を越えたことを検出する(ステップ2501)と、検出された記憶装置に含まれている部分ファイルを、部分ファイル管理テーブル1301

(図13)を参照して探し出す。そして、探し出した部分ファイルの「単位時間当たりのアクセス情報」を、アクセス情報テーブル1201から獲得する。獲得した「単位時間当たりのアクセス情報」の「アクセス回数」を比較し、最も大きな「アクセス回数」になっている部分ファイルを、コピー元部分ファイルとして選択する(ステップ2502)。ここでは、例えば、部分ファイルA1が選択されたとする。

【0198】次に、外部負荷情報テーブル901(図9)に基づいて、記憶装置の記憶部の「残容量」が十分あり、「負荷」が所定の値より低い記憶装置を持つ複数のサーバ計算機を選択する。そして、選択されたサーバ計算機に対して部分ファイルをコピーできるかどうかを確認し、コピー可能なサーバ計算機を、コピー先サーバ計算機として決定する(ステップ2502)。

【0199】ここで、例えば、サーバ計算機B106の記憶装置識別子DiskID1で示される記憶部を有する記憶装置120と、サーバ計算機C107の記憶装置識別子DiskID2で示される記憶部を有する記憶装置125が選択されたとする。

【0200】次に、上述のステップ2502で得られたコピー元部分ファイルとコピー先サーバ計算機の情報に基づいて、部分ファイルのコピーを行ない(ステップ2503)、ステップ2501の監視処理を続行する。

【0201】上述の例において、ステップ2502で、コピー元部分ファイルとして部分ファイルA1(126a)、コピー先計算機としてサーバ計算機B2006とサーバ計算機C2007が選択されたので、コピー元のサーバ計算機A2005の分散ファイルコピー部2031は部分ファイル管理部111を介して、記憶装置115から部分ファイルA1(126a)を読み出し、ネットワークインタフェース113を介してネットワーク101へマルチキャストで送信する。また、同時に部分ファイルA1(126a)の「オリジナル所在地」(図13)に関する情報も送信する。

【0202】一方、コピー先のサーバ計算機B2006と計算機C2007では、分散ファイルコピー部2032、2033がコピー元のサーバ計算機A2005より送信された部分ファイルA1(126a)を、ネットワークインタフェース118、123を介して受信し、部分ファイル管理部116、121を介して、記憶装置1

20、125に書き込む。また、部分ファイルA1(126a)の「オリジナル所在地」(図13)も受信して、それぞれの部分ファイル管理テーブル1402、1403(図14)に登録する。その後、コピー先のサーバ計算機B2006とサーバ計算機C2007は、それぞれコピー元のサーバ計算機と「オリジナル所在地」

(図13)に示されているサーバ計算機、すなわち、この場合には、両方ともサーバ計算機A2005であるため、両者ともこのサーバ計算機A2005に対して、部分ファイルA1(126a)のコピーが完了したことを通知する。コピー元のサーバ計算機と「オリジナル所在地」(図13)に示されるサーバ計算機、すなわち、サーバ計算機A2005では部分ファイル管理テーブル1401(図14)の部分ファイルA1(126a)の情報を書き換える。

【0203】以上のように、本実施の形態では、分散ファイルコピー部2031、2032、2033が部分ファイルを他のサーバ計算機へコピーする際に、複数のコピー先のサーバ計算機の候補を選択し、選択された複数のサーバ計算機へマルチキャスト通信により同時に部分ファイルをコピーするため、部分ファイルのコピーの際の通信量を削減できる。

【0204】(実施の形態7)図26は、本発明における分散ファイル管理システムの他の実施の形態の一例を示す構成図である。ここで、図26では、図1と同一の構成のものには同一の符号を付している。図26において、この分散ファイル管理システムは、パーソナルコンピュータやワークステーションなどのサーバ計算機及びパーソナルコンピュータやワークステーションなどの複数のクライアント計算機から成るクライアント計算機群を備えた複数の計算機サイトA2602、計算機サイトB2603、及び計算機サイトC2604と、計算機サイトA2602、計算機サイトB2603、及び計算機サイトC2604を相互に接続するローカルエリアネットワークやワイドエリアネットワークなどのネットワーク101とを備えている。

【0205】ここで、計算機サイトA2602は、パーソナルコンピュータやワークステーションなどの複数のサーバ計算機(図26においては、「サーバ計算機A2605」のみ示す)と、パーソナルコンピュータやワークステーションなどのクライアント計算機1~n(108-1~108-n)から成るクライアント計算機群A108とを備えている。この計算機サイトA2602は、複数のサーバ計算機(図26においては、「サーバ計算機A2605」のみ示す)とクライアント計算機群A108とをイーサネットなどの内部ネットワーク131で接続しており、例えば、インターネットドメインになっている。

【0206】また、計算機サイトA2602と同様に、計算機サイトB2603は、複数のサーバ計算機(図2

6においては、「サーバ計算機B2606」のみ示す)と、複数のクライアント計算機から成るクライアント計算機群B2609とを備え、計算機サイトC2604は、複数のサーバ計算機(図26においては、「サーバ計算機C2607」のみ示す)と、複数のクライアント計算機から成るクライアント計算機群C2610とを備えている。さらに、これらの計算機サイトB2603及び計算機サイトC2604は、計算機サイトA2602と同様に、複数のサーバ計算機(図26においては、「サーバ計算機B2606」及び「サーバ計算機C2607」のみ示す)と、クライアント計算機群B109及びクライアント計算機群C110とを、それぞれ内部ネットワーク132及び内部ネットワーク133で接続しており、例えば、インターネットドメインになっている。

【0207】サーバ計算機A2605は、分散ファイルの部分ファイルを記録するハードディスクなどの記憶装置115と、イーサネットなどの内部ネットワーク131へ接続するためのネットワークインタフェース113と、部分ファイルを記録している記憶装置115への書き込みや読み出しを制御する部分ファイル管理部111と、記憶装置115に対する負荷や記憶装置115の残り容量、及びネットワークインタフェース113に対する負荷を監視し、これらの負荷や容量に関する情報を保持する状態管理部114と、部分ファイル管理部111、状態管理部114、及びネットワークインタフェース113に接続された分散ファイル管理部2612とによって構成されている。

【0208】この分散ファイル管理部2612は、部分ファイルの書き込みや読み出しを部分ファイル管理部111に指示する。また、分散ファイル管理部2612は、分散ファイルを作成する場合には、状態管理部114から得られる情報に基づいて、分散ファイルを複数の部分ファイルに分割し、各部分ファイルを配置(記録)するサーバ計算機を決定する。また、以前に作成された分散ファイルを参照または更新する場合には、該当する分散ファイルの部分ファイルが存在する(記録されている)サーバ計算機を検出する。ここで、分散ファイル管理部2612は、クライアント計算機からの情報または分散ファイルに記録されるデータの種別に応じて、分散ファイルを部分ファイルへ分割する際の部分ファイルのサイズを決定する部分ファイルサイズ決定部2631を備えている。

【0209】サーバ計算機B2606及びサーバ計算機C2607は、サーバ計算機A2605と同様の構成になっている。すなわち、サーバ計算機B2606は、記憶装置120と、ネットワークインタフェース118と、部分ファイル管理部116と、状態管理部119と、分散ファイル管理部2617とによって構成されている。また、サーバ計算機C2607は、記憶装置12

5と、ネットワークインタフェース123と、部分ファイル管理部121と、状態管理部124と、分散ファイル管理部2622とによって構成されている。また、分散ファイル管理部2617、2622は、それぞれ部分ファイルサイズ決定部2632、2633を備えている。

【0210】ここで、図26に示した分散ファイル管理システムと図に示した分散ファイル管理システムとの相違点は、図26の分散ファイル管理システムにおいて、分散ファイル管理部2612、2617、2622に、クライアント計算機からの情報または分散ファイルに記録されるデータの種別に応じて、分散ファイルを部分ファイルへ分割する際の部分ファイルのサイズを決定する部分ファイルサイズ決定部2631、2632、2633を備えている点である。

【0211】以上のように構成された分散ファイル管理システムの部分ファイルサイズ決定部2631、2632、2633の動作について以下に説明する。

【0212】図3において、例えば、クライアント計算機1(108-1)がサーバ計算機A2605に対して、分散ファイルAの作成要求を行なった時、ステップ302の処理で、分散ファイル管理部2612の部分ファイルサイズ決定部2631が、部分ファイルを割り当てる時のサイズを決定する。部分ファイルサイズ決定部2631が、部分ファイルのサイズを決定する際には、分散ファイルに記録されるデータの種別(例えば、M-JPEG、MPEG1、MPEG2など)やクライアント計算機1(108-1)からの指示によって決めるようにするとよい。

【0213】以上のように、本実施の形態の分散ファイル管理システムにおいて、分散ファイル管理部2612、2617、2622の部分ファイルサイズ決定部2631、2632、2633が、クライアント計算機からの情報や分散ファイルに記録されるデータの種別などによって、分散ファイルを部分ファイルへ分割する際の部分ファイルのサイズを決定することにより、分散ファイルを構成する部分ファイルのサイズを適宜変更することができる。これにより、論理的や内容的に関連のあるデータ、例えば、画像1フレーム分のデータなどを複数の部分ファイルに分割してしまうことを防止することができる。

【0214】(実施の形態8)次に、上述した複数のサーバ計算機が有する分散ファイル管理部及び状態管理部を1つのサーバ計算機にまとめ、該サーバ計算機で集中して管理する場合について説明する。

【0215】図27は、本発明における第1の実施の形態に示した分散ファイル管理システムにおいて、分散ファイル管理部及び状態管理部を1つのサーバ計算機にまとめた場合の分散管理システムの構成の一例を示している。図27において、図1と同一の構成のものには同一

の符号を付している。

【0216】図27に示した分散ファイル管理システムは、パーソナルコンピュータやワークステーションなどのサーバ計算機及びパーソナルコンピュータやワークステーションなどの複数のクライアント計算機から成るクライアント計算機群を備えた複数の計算機サイトA2702、計算機サイトB2703、及び計算機サイトC2704と、他のサーバ計算機上に配置されている分散ファイルを集中的に管理する管理サーバ計算機X2711を備えた計算機サイトX2710と、計算機サイトA2702、計算機サイトB2703、計算機サイトC2704、及び計算機サイトX2710を相互に接続するローカルエリアネットワークやワイドエリアネットワークなどのネットワーク101とを備えている。

【0217】ここで、計算機サイトA2702は、パーソナルコンピュータやワークステーションなどの複数のサーバ計算機（図27においては、「サーバ計算機A2705」のみ示す）と、パーソナルコンピュータやワークステーションなどのクライアント計算機1～n（108-1～108-n）から成るクライアント計算機群A108とを備えている。この計算機サイトA2702は、複数のサーバ計算機（図27においては、「サーバ計算機A2705」のみ示す）とクライアント計算機群A108とをイーサネットなどの内部ネットワーク131で接続しており、例えば、インターネットドメインになっている。

【0218】また、計算機サイトA2702と同様に、計算機サイトB2703は、複数のサーバ計算機（図27においては、「サーバ計算機B2706」のみ示す）と、複数のクライアント計算機から成るクライアント計算機群B109とを備え、計算機サイトC2704は、複数のサーバ計算機（図27においては、「サーバ計算機C2707」のみ示す）と、複数のクライアント計算機から成るクライアント計算機群C110とを備えている。さらに、これらの計算機サイトB2703及び計算機サイトC2704は、計算機サイトA2702と同様に、複数のサーバ計算機（図27においては、「サーバ計算機B2706」及び「サーバ計算機C2707」のみ示す）と、クライアント計算機群B109及びクライアント計算機群C110とを、それぞれ内部ネットワーク132及び内部ネットワーク133で接続しており、例えば、インターネットドメインになっている。

【0219】サーバ計算機A2705は、分散ファイルの部分ファイルを記録するハードディスクなどの記憶装置115と、イーサネットなどの内部ネットワーク131へ接続するためのネットワークインタフェース113と、部分ファイルを記録している記憶装置115への書き込みや読み出しを制御する部分ファイル管理部111とによって構成されている。

【0220】サーバ計算機B2706及びサーバ計算機

C2707は、サーバ計算機A2705と同様の構成になっている。すなわち、サーバ計算機B2706は、記憶装置120と、ネットワークインタフェース118と、部分ファイル管理部116とによって構成されている。また、サーバ計算機C2707は、記憶装置125と、ネットワークインタフェース123と、部分ファイル管理部121とによって構成されている。

【0221】管理サーバ計算機X2711は、イーサネットなどの内部ネットワーク134に接続するためのネットワークインタフェース2713と、各サーバ計算機の記憶装置の負荷や残り容量、ネットワークインタフェースの負荷を監視し、負荷に関する情報を保持する状態管理部2714と、部分ファイルの書き込みや読み出しを各サーバ計算機の部分ファイル管理部111、116、121に指示したり、分散ファイルを作成する際に、状態管理部2714からの情報に基づいて分散ファイルを複数の部分ファイルに分割し、各部分ファイルを配置するサーバ計算機を決定することによって部分ファイルの作成を行ない、また、分散ファイルを参照または更新する場合には、参照または更新される当該分散ファイルの部分ファイルが在左するサーバ計算機を検出して部分ファイルの参照または更新を行なう分散ファイル管理部2712によって構成されている。

【0222】図27においては、各分散ファイルA、B、Cが作成された後の状態を示している。すなわち、サーバ計算機A2705の記憶装置115には、分散ファイルAの部分ファイルA1（126a）と分散ファイルBの部分ファイルB1（126b）とが記録されている。また、サーバ計算機B2706の記憶装置120には、分散ファイルCの部分ファイルC1（126e）と分散ファイルCの部分ファイルC2（126f）とが記録されている。また、サーバ計算機C2707の記憶装置125には、分散ファイルAの部分ファイルA2（126c）と分散ファイルAの部分ファイルA3（126d）とが記録されている。

【0223】次に、以上のように構成された分散ファイル管理システムの動作について説明する。以下においては、クライアント計算機群A108のクライアント計算機1（108-1）からサーバ計算機A2705に対して分散ファイルAの作成要求が発行され、図27に示したような部分ファイルA1～A3が作成される場合の分散処理を例にして説明する。ここで、図27に示した記憶装置115、120、125は、それぞれ複数の記憶部または記憶領域（以下、単に「記憶部」ともいう）を有するものとする。これらの複数の記憶部は、物理的に1つの記録媒体であってもよく、また、複数の記録媒体であってもよい。

【0224】図27において、まず、クライアント計算機1（108-1）から計算機サイトX2710の管理サーバ計算機X2711に分散ファイルAの作成要求が

発行される。この分散ファイルAの作成要求は、計算機サイトA2702の内部ネットワーク131、ネットワーク101、計算機サイトX2710の内部ネットワーク134、及び管理サーバ計算機A2711のネットワークインタフェース2713を介して、分散ファイル管理部2712によって受け取られる。

【0225】図28は、分散ファイルの作成要求を受け取った場合の分散ファイル管理部2712の動作アルゴリズムを示すフローチャートである。以下、図27及び図28を用いて、分散ファイル管理部2712の詳細な動作を説明する。

【0226】図28において、まず、分散ファイル管理部2712は、状態管理部2714の管理している負荷情報を参照する（ステップ2801）。

【0227】状態管理部2714では、例えば、図4のような負荷情報テーブル401を管理している。図4では、サーバ計算機A2705に関する情報について示しているが、状態管理部2714では、図4に示したような負荷情報テーブル401を各サーバ計算機毎に準備し管理している。図4において、負荷情報テーブル401は、記憶装置負荷情報テーブル402とネットワーク負荷情報テーブル403からなる。記憶装置負荷情報テーブル402は、サーバ計算機に接続されている記憶装置の各記憶部を識別するための「記憶装置識別子」と、記憶装置の各記憶部の「負荷」と、記憶装置の各記憶部の「残容量」の情報で構成されている。記憶装置の各記憶部の「負荷」は、記憶部の最大転送レートのうち、何【％】を使用しているかで表示している。ネットワーク負荷情報テーブル403は、それぞれのサーバ計算機のネットワークインタフェースを介して送信するデータがどのサイトに向けてのものであり、どの程度の帯域幅を使用して送信されているか、また、受信しているデータがどのサイトから送られて来たものであり、どの程度の帯域幅を使用して受信しているかを表している。また、「送出元サイト」がデータの送出元の計算機サイトを示し、「送出先サイト」がデータの送出先の計算機サイトを示し、「使用帯域幅」が送出元の計算機サイトと送出先の計算機サイトの間で使用されている通信帯域幅を示している。

【0228】例えば、分散ファイルAを作成する場合、分散ファイル管理部2712は、サーバ計算機A2705の記憶装置115の記憶装置識別子がDiskID1で示される記憶部について、その「負荷」が20【％】で、その「残容量」が10【Mbytes】であるという情報を得ることができる。

【0229】次に、分散ファイル管理部2712は、状態管理部2714から得られた「記憶装置負荷情報」に基づいて、各サーバ計算機に接続されている記憶装置の各記憶部の中から、「残容量」の値が所定の値より大きく、「負荷」の値が所定の値、例えば、80【％】（こ

の値は、システムや他の装置の構成に応じて決定される）より低い記憶部を有する記憶装置を選択する。そして、この記憶装置の記憶部に部分ファイルを順番に割り当てる。このとき、部分ファイルのサイズを全てのサーバ計算機で同一の固定長とするとよい。この部分ファイルの割り当て処理において、全ての部分ファイルを割り当てることができたかどうかを検知する（ステップ2802）。

【0230】ここで、分散ファイルAを作成する場合、図27においては、部分ファイルA1（126a）をサーバ計算機A2705に割り当て、部分ファイルA2（126c）及び部分ファイルA3（126d）をサーバ計算機C2704に割り当てることになる。

【0231】全ての部分ファイルを割り当てることができた場合には、分散ファイル管理部2712は、分散ファイルを管理するための情報を、例えば、上述したような図5の分散ファイル管理テーブル501と図6の部分ファイル管理テーブル601に登録する（ステップ2803）。

【0232】上述の例においては、図5に示すように、分散ファイルAは、部分ファイルA1（126a）、部分ファイルA2（126c）、及び部分ファイルA3（126d）から構成されている。

【0233】また、図6において、部分ファイルA1（126a）の所在地が、「file:///siteA/serverA/DiskID1/（計算機サイトA102のサーバ計算機A105の記憶装置識別子DiskID1）」であり、部分ファイルA2（126c）の所在地が、「file:///siteC/serverC/DiskID2/（計算機サイトC104のサーバ計算機C107の記憶装置識別子DiskID2）」であり、部分ファイルA3（126d）の所在地が、「file:///siteC/serverC/DiskID2/（計算機サイトC104のサーバC107の記憶装置DiskID2）」であることを表している。

【0234】次に、分散ファイル管理部2712は、ステップ2803で登録した部分ファイルを該当する各サーバ計算機上に作成するため、作成を行なうサーバ計算機の部分ファイル管理部に部分ファイルの作成要求を行う。この作成要求と同時に、分散ファイルの作成要求を行なったクライアント計算機1（108-1）に対して、クライアント計算機1（108-1）から直接部分ファイルの作成を行なうサーバ計算機にデータを送信するように指示する。部分ファイルの作成を要求されたサーバ計算機では、部分ファイル管理部によって記憶装置の各記憶部にクライアント計算機1（108-1）からのデータの書き込みを行なう。分散ファイル管理部2712は、全ての部分ファイルの作成が終るまでこの処理を繰り返す（ステップ2804）。

【0235】ここで、分散ファイルAの部分ファイルA

1 (126 a) の作成の場合、サーバ計算機A2705の部分ファイル管理部111が、クライアント計算機1 (108-1) からのデータを記憶装置115の所定の記憶部に書き込む。また、サーバ計算機C2707の部分ファイル管理部121は、クライアント計算機1 (108-1) からのデータを記憶装置125の所定の記憶部に書き込んで、部分ファイルA2 (126 c) 及び部分ファイルA3 (126 d) を作成する。

【0236】一方、ステップ2802で、全ての部分ファイルを割り当てることができなかった場合には、分散ファイル管理部2712は、分散ファイルの作成要求を行なったクライアント計算機1 (108-1) に対して、分散ファイルの作成処理が失敗したことを通知する (ステップ2805)。

【0237】次に、クライアント計算機から管理サーバ計算機に対して分散ファイルの参照または更新の要求 (以下、「参照/更新要求」ともいう) が発行された場合について説明する。また、以下の説明において、クライアント計算機群A108内のクライアント計算機1 (108-1) から管理サーバ計算機X2711に対して分散ファイルAの参照/更新要求が発行された場合を例にして述べる。

【0238】まず、クライアント計算機1 (108-1) によって発行された分散ファイルAに対する参照/更新要求は、管理サーバ計算機X2711において、ネットワークインタフェース2713を介して、分散ファイル管理部2712によって受信される。

【0239】図29は、分散ファイル管理部が分散ファイルの参照/更新要求を受け取った場合の動作アルゴリズムを示すフローチャートである。以下、図29を用いて、分散ファイル管理部2712の動作を説明する。

【0240】まず、分散ファイル管理部2712は、クライアント計算機1 (108-1) からの分散ファイルAの参照/更新要求に応じて、分散ファイル管理テーブル501 (図5) と部分ファイル管理テーブル601 (図6) から、更新または参照する部分ファイルと、その部分ファイルの所在地を求める (ステップ2901)。

【0241】ここで、分散ファイルAの参照/更新要求の場合、分散ファイル管理テーブル501 (図5) から、分散ファイルAは、部分ファイルA1 (126 a)、部分ファイルA2 (126 c) 及び部分ファイルA3 (126 d) により構成されていることがわかる。また、部分ファイル管理テーブル601 (図6) によって、部分ファイルA1 (126 a) は、「file://siteA/serverA/DiskID1/」で示される記憶装置115の記憶部に存在し、部分ファイルA2 (126 c) は、「file://siteC/serverC/DiskID2/」で示される記憶装置125の記憶部に存在し、部分ファイルA3 (126

d) は、「file://siteC/serverC/DiskID2/」で示される記憶装置125の記憶部に存在することが解る。

【0242】分散ファイル管理部2712は、クライアント計算機1 (108-1) からの参照/更新要求に対応する部分ファイルを保持するサーバ計算機に対して、当該部分ファイルの参照/更新要求を行う。この要求と同時に、分散ファイル管理部2712は、クライアント計算機1 (108-1) に対して、クライアント計算機1 (108-1) が参照または更新を行う部分ファイルの存在するサーバ計算機に直接、参照/更新要求を行うように指示する。各サーバ計算機の部分ファイル管理部は、分散ファイル管理部2712からの要求に応じ、クライアント計算機1 (108-1) からの参照/更新要求に基づいて、記憶装置に存在する部分ファイルの読み出し (参照)、または記憶装置への部分ファイルの書き込み (更新) を行なう (ステップ2902)。

【0243】ここで、分散ファイルAの場合、部分ファイルA1 (126 a) は、計算機サイトA2702のサーバ計算機A2705に、部分ファイルA2 (126 c) は、計算機サイトC2704のサーバ計算機C2707に、部分ファイルA3 (126 d) は、計算機サイトC2704のサーバ計算機C2707に存在している。したがって、部分ファイルA1 (126 a) に対する参照/更新要求の処理は、分散ファイル管理部2712からの要求に応じて、クライアント計算機1 (108-1) と部分ファイル管理部111との間で直接行なわれる。一方、部分ファイルA2 (126 c) と部分ファイルA3 (126 d) に対する参照/更新要求の処理は、分散ファイル管理部2712からの要求に応じて、クライアント計算機1 (108-1) と、サーバ計算機C2707との間で直接行なわれる。

【0244】以上のように、本実施の形態の分散ファイル管理システムによれば、第1の実施の形態に示した効果に加え、分散ファイルの管理とシステムの状態の管理を集中して行うため、重複した管理部を複数持つ必要がなく、システム構成を簡略することができ、またコストの軽減を図ることができる。

【0245】尚、上述した図27の分散ファイル管理システムにおいては、第1の実施の形態で示した分散ファイル管理システムの管理部を、1つの管理サーバ計算機に集中した構成として説明したが、第2～第7で示した実施の形態の分散ファイル管理システムにも適用することができる。

【0246】また、上述した図27の分散ファイル管理システムにおいて、分散ファイル管理システムの管理部を、1つの管理サーバ計算機に集中した構成として説明したが、所定のグループのサーバ計算機毎や所定のグループの計算機サイト毎に管理サーバ計算機を設けるようにしてもよい。このようにすると、大規模なシステムに

における管理サーバ計算機への負荷の集中を防止することができる。

【0247】また、上述した実施の形態において、部分ファイルが複数のサーバ計算機上に存在する場合の部分ファイルの選択においては、単にサーバ計算機の「負荷」の小さい順に選択するだけでなく、サーバ計算機を所定の規則に基づいて順番に使用するようにしてもよく、また、所定の閾値以下の「負荷」を有するサーバ計算機をランダムに選択するようにしてもよい。

【0248】

【発明の効果】以上のように、本発明の分散ファイル管理装置及び分散ファイル管理システムによれば、クライアント計算機からサーバ計算機に対する分散ファイルの作成、参照、または更新の要求に応じて、要求されたサーバ計算機または管理サーバ計算機の各々の管理部で、各サーバ計算機の負荷情報に基づいて部分ファイルの配置を決定するため、特定のサーバ計算機への負荷の集中を回避することができるようになった。

【0249】また、他のサーバ計算機へ負荷情報を通知し、また、他のサーバ計算機から通知された外部負荷情報を保持して、他のサーバ計算機の負荷情報に基づいて、部分ファイルの配置を決定するため、特定のサーバ計算機への負荷の集中を回避することができるようになった。

【0250】また、部分ファイル毎のアクセス情報、負荷情報及び外部負荷情報に基づいて、移動させる部分ファイルを決定し、他のサーバ計算機へ部分ファイルを移動するため、特定のサーバ計算機の記憶装置への負荷の集中や、各サーバ計算機の記憶装置の容量の不均衡を回避することができるようになった。

【0251】また、部分ファイル毎のアクセス状況、負荷情報及び外部負荷情報に基づいて、コピーする部分ファイルを決定し、他のサーバ計算機に部分ファイルをコピーするため、特定のサーバ計算機の記憶装置への負荷の集中を回避することができるようになった。

【0252】また、クライアント計算機からの情報や分散ファイルに記録されるデータの種類によって、分散ファイルを分割して作成する部分ファイルのサイズを決定するため、分散ファイルを構成する部分ファイルのサイズを適宜変更することができ、内容的、論理的に関連のあるデータ、例えば、画像1フレーム分のデータなどを複数の部分ファイルに分けて記録することを防止することができるようになった。

【0253】また、分散ファイルの部分ファイルを管理する各管理部を1つまたは複数の管理用サーバ計算機に集中させることにより、リソースの重複を最小限に抑えることができるため、コストの増加を抑えることができるようになった。

【図面の簡単な説明】

【図1】本発明の分散ファイル管理システムを示すブ

ック図である。

【図2】本発明における分散ファイルの構成を示す図である。

【図3】本発明の分散ファイル管理部の分散ファイル作成アルゴリズムを示すフローチャートである。

【図4】本発明における負荷情報テーブルの一例を示す図である。

【図5】本発明の分散ファイル管理テーブルの一例を示す図である。

【図6】本発明の部分ファイル管理テーブルの一例を示す図である。

【図7】本発明の分散ファイル管理部の分散ファイルの参照／更新アルゴリズムを示すフローチャートである。

【図8】本発明の分散ファイル管理システムを示すブロック図である。

【図9】本発明における外部負荷情報テーブルの一例を示す図である。

【図10】本発明における分散ファイル管理部の分散ファイル作成アルゴリズムを示すフローチャートである。

【図11】本発明の分散ファイル管理システムを示すブロック図である。

【図12】本発明における部分ファイルアクセス情報テーブルの一例を示す図である。

【図13】本発明における部分ファイル管理テーブルの一例を示す図である。

【図14】本発明における部分ファイル管理テーブルの一例を示す図である。

【図15】本発明における分散ファイル移動部の動作アルゴリズムを示すフローチャートである。

【図16】本発明における部分ファイル管理テーブルの一例を示す図である。

【図17】本発明における部分ファイル管理テーブルの一例を示す図である。

【図18】本発明における分散ファイル移動部の動作アルゴリズムを示すフローチャートである。

【図19】本発明における接続情報テーブルの一例を示す図である。

【図20】本発明の分散ファイル管理システムを示すブロック図である。

【図21】本発明における分散ファイルコピー部の動作アルゴリズムを示すフローチャートである。

【図22】本発明における部分ファイル管理テーブルの一例を示す図である。

【図23】本発明における部分ファイル管理テーブルの一例を示す図である。

【図24】本発明における分散ファイルコピー部の動作アルゴリズムを示すフローチャートである。

【図25】本発明における分散ファイルコピー部の動作アルゴリズムを示すフローチャートである。

【図26】本発明の分散ファイル管理システムを示すブ

ロック図である。

【図27】本発明の分散ファイル管理システムを示すブロック図である。

【図28】本発明の分散ファイル管理部の分散ファイル作成アルゴリズムを示すフローチャートである。

【図29】本発明の分散ファイル管理部の分散ファイルの参照／更新アルゴリズムを示すフローチャートである。

【図30】従来の分散ファイル管理装置を示すブロック図である。

【符号の説明】

101 ネットワーク
102～104、802～804、1102～1104、2002～2004、2602～2604、2702～2704、2710 計算機サイト
105～107、805～807、1105～1107、2005～2007、2605～2607、2705～2707 サーバ計算機
108、109、110 クライアント計算機群
108-1～108-n クライアント計算機
111、116、121 部分ファイル管理部
112、117、122、1112、1117、1122、2012、2017、2022、2612、2617、2622、2712 分散ファイル管理部

113、118、123、2713 ネットワークインタフェース

114、119、124、814、819、824、2714 状態管理部

115、120、125 記憶装置

126a～126f、202-1～202-n 部分ファイル

131～133 内部ネットワーク

401 負荷情報テーブル

402 記憶装置負荷情報テーブル

403 ネットワーク負荷情報テーブル

501 分散ファイル管理テーブル

601、1301、1401～1403、1601～1603、1701～1703、2201～2203、2301～2303 部分ファイル管理テーブル

811、812、813 外部状態管理部

901 外部負荷情報テーブル

1131、1132、1133 分散ファイル移動部

1201 アクセス情報テーブル

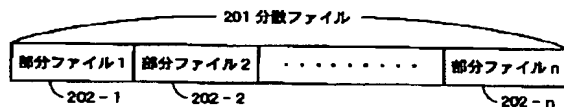
1901 接続情報テーブル

2031、2032、2033 分散ファイルコピー部

2631、2632、2633 部分ファイルサイズ決定部

2711 管理サーバ計算機

【図2】



【図5】

分散ファイル識別子	部分ファイル識別子リスト
A	A1, A2, A3
B	B1
C	C1, C2
⋮	⋮
⋮	⋮

【図4】

記憶装置負荷情報		
記憶装置識別子	負荷	残容量
Disk ID 1	20 %	10 Mbytes
Disk ID 2	30 %	200 Mbytes
⋮	⋮	⋮
⋮	⋮	⋮

ネットワーク負荷情報		
リンク		使用通信帯域幅
送出元サイト	送出先サイト	
site A	site B	30 Mbps
site A	site C	50 Mbps
⋮	⋮	⋮
⋮	⋮	⋮

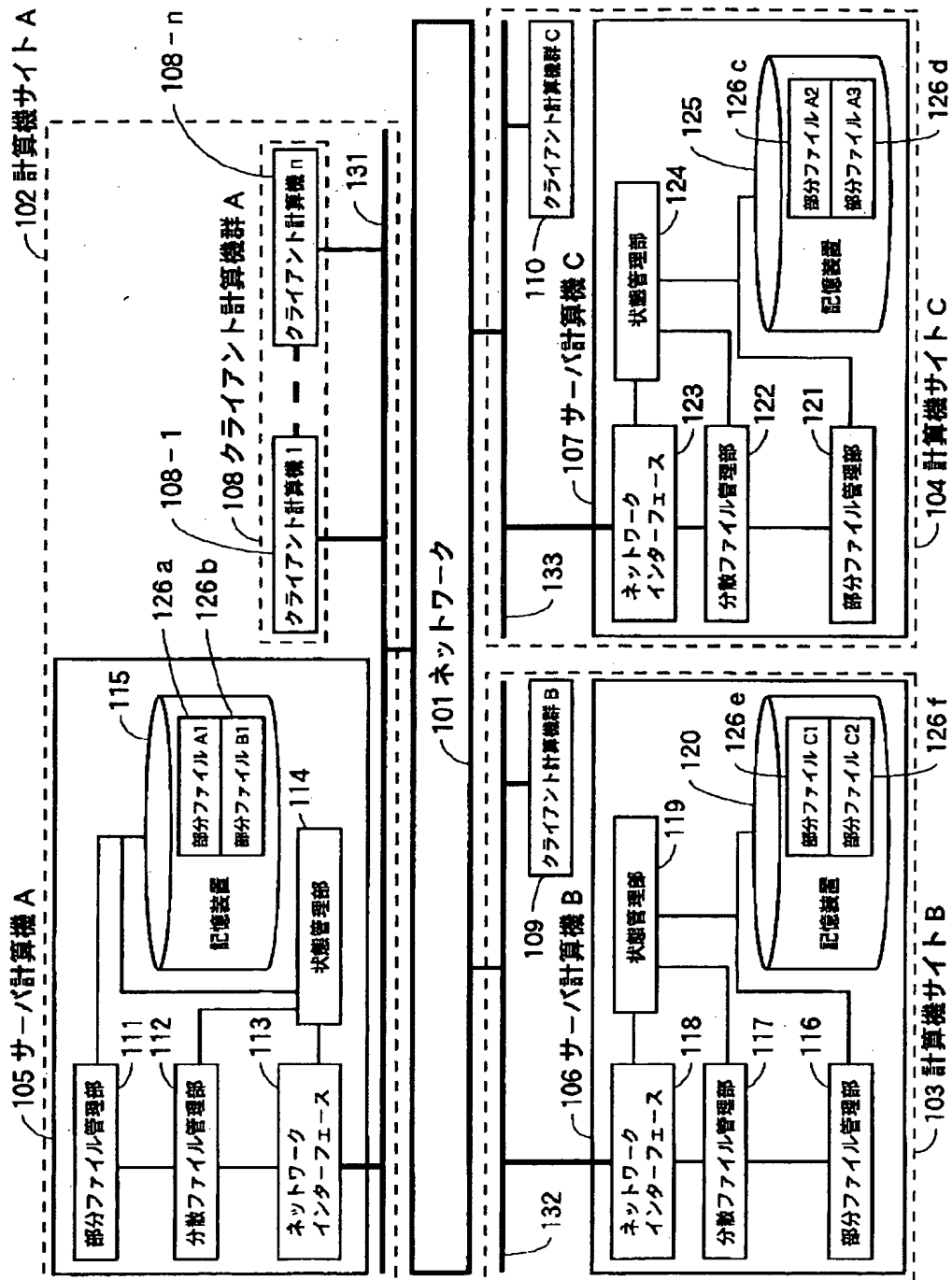
【図6】

【図9】

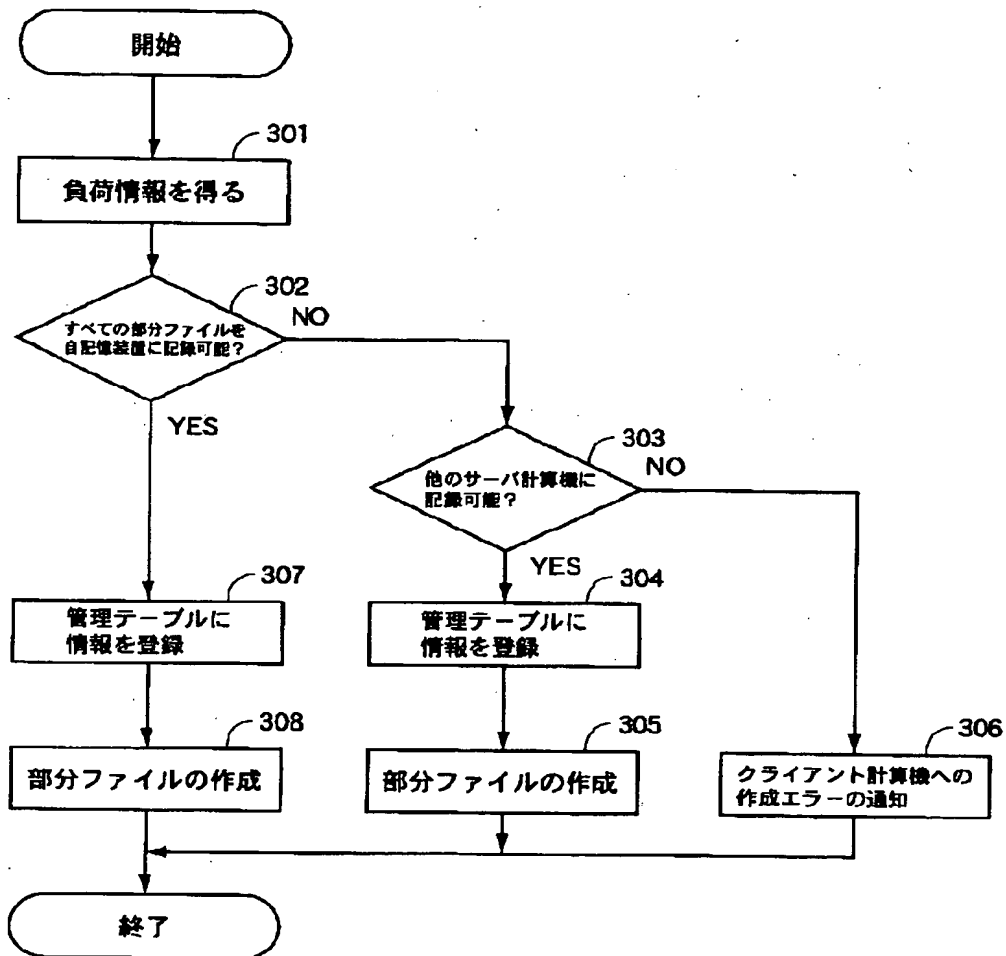
サーバ計算機所在地	記憶装置負荷情報		
	記憶装置識別子	負荷	残容量
site B/server B	Disk ID 1	49 %	1000 Mbytes
site C/server C	Disk ID 1	30 %	3000 Mbytes
⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮

部分ファイル識別子	所在地
A1	file://site A/server A/Disk ID 1/
A2	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/
B1	file://site A/server A/Disk ID 1/
C1	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/
⋮	⋮
⋮	⋮

【図1】



【図3】



【図12】

部分ファイル識別子	単位時間あたりのアクセス回数	
	アクセス元サイト識別子	アクセス回数
A1	site B	10
A2	site A	2
B1	site C	5
⋮	⋮	⋮
⋮	⋮	⋮

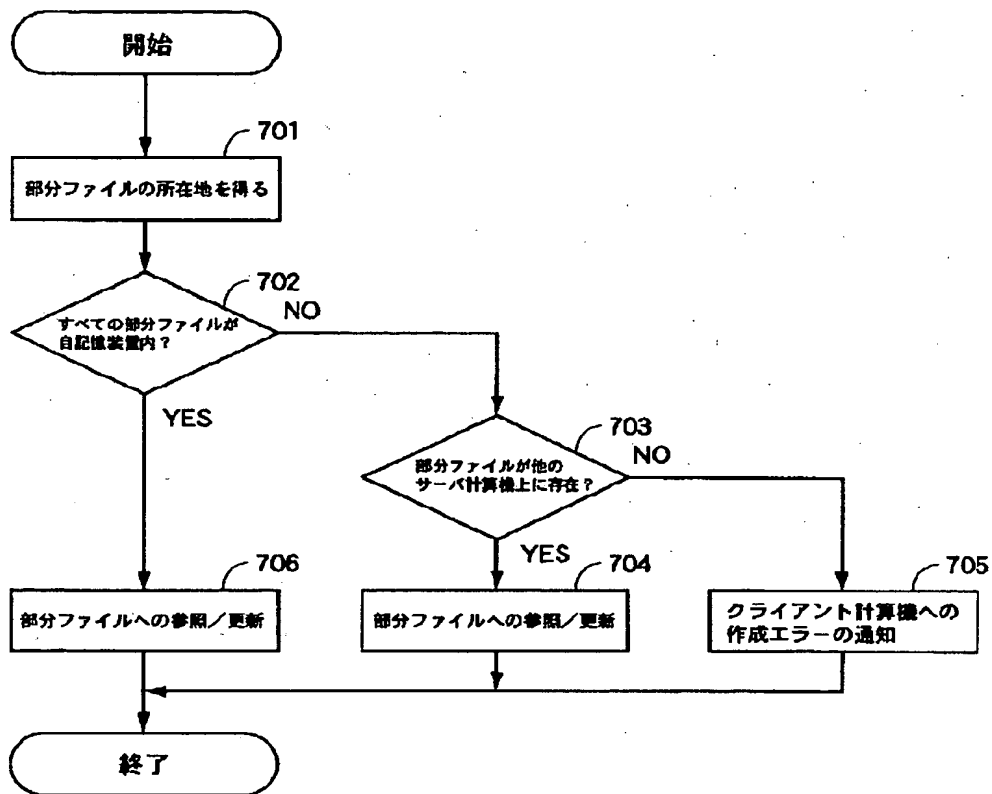
【図13】

部分ファイル識別子	所在地	オリジナル所在地
A1	file://site A/server A/Disk ID 1/	file://site A/server A/Disk ID 1/
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
B1	file://site A/server A/Disk ID 2/	file://site A/server A/Disk ID 2/
C1	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮
⋮	⋮	⋮

【図19】

送元サイト	送先サイト	通信時間
site A	site B	10 ms
site B	site A	20 ms
⋮	⋮	⋮
⋮	⋮	⋮

【図7】



【図14】

(A)

部分ファイル識別子	所在地	オリジナル所在地
A1	file://site A/server A/Disk ID 1/	file://site A/server A/Disk ID 1/
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
B1	file://site A/server A/Disk ID 1/	file://site A/server A/Disk ID 1/
C1	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮

(B)

部分ファイル識別子	所在地	オリジナル所在地
C1	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮

(C)

部分ファイル識別子	所在地	オリジナル所在地
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
⋮	⋮	⋮

【図15】

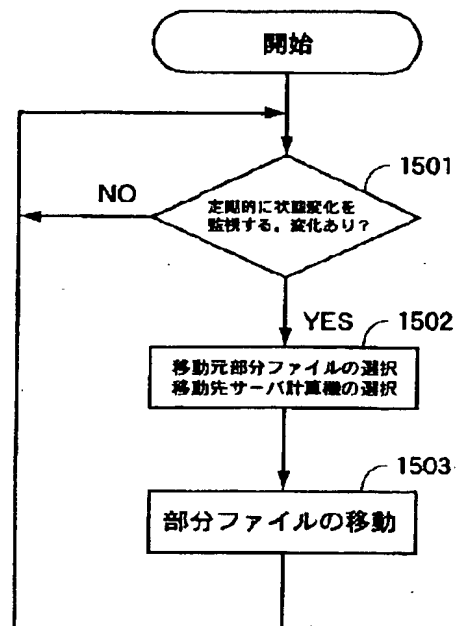
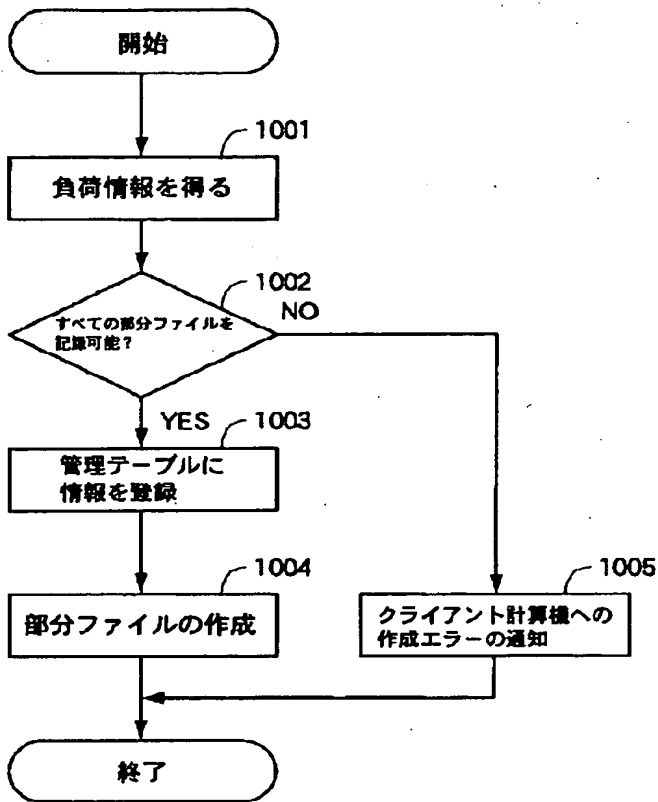


Figure 1 is a block diagram of a system architecture. At the center is a horizontal bar labeled "101 ネットワーク" (Network). To the left of the network are three main computer groups, each enclosed in a dashed box. The top group is labeled "805 サーバ計算機 A" (Server Computer A) and contains a "805 サーバ計算機 A" (Server Computer A) and multiple "108 クライアント計算機群 A" (Client Computer Group A) labeled 108-1, 108-2, ..., 108-n. The middle group is labeled "806 サーバ計算機 B" (Server Computer B) and contains a "806 サーバ計算機 B" (Server Computer B) and multiple "109 クライアント計算機群 B" (Client Computer Group B) labeled 109-1, 109-2, ..., 109-n. The bottom group is labeled "807 サーバ計算機 C" (Server Computer C) and contains a "807 サーバ計算機 C" (Server Computer C) and multiple "110 クライアント計算機群 C" (Client Computer Group C) labeled 110-1, 110-2, ..., 110-n. Each server computer group has internal components: a "部分ファイル管理部" (Partial File Management Unit), a "分散ファイル管理部" (Distributed File Management Unit), a "ネットワークインタフェース" (Network Interface), a "状態管理部" (Status Management Unit), an "外部状態管理部" (External Status Management Unit), and a "記憶装置" (Storage Device). The client computer groups also have internal components: a "部分ファイル管理部" (Partial File Management Unit), a "分散ファイル管理部" (Distributed File Management Unit), a "ネットワークインタフェース" (Network Interface), a "状態管理部" (Status Management Unit), an "外部状態管理部" (External Status Management Unit), and a "記憶装置" (Storage Device). The diagram shows connections between the server and client groups and within each group.

【図10】



【図16】

(A)

部分ファイル識別子	所在地	オリジナル所在地
A1	file://site C/server C/Disk ID 2/	file://site A/server A/Disk ID 1/
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
B1	file://site A/server A/Disk ID 1/	file://site A/server A/Disk ID 1/
C1	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮
⋮	⋮	⋮

(B)

部分ファイル識別子	所在地	オリジナル所在地
C1	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮
⋮	⋮	⋮

(C)

部分ファイル識別子	所在地	オリジナル所在地
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A1	file://site C/server C/Disk ID 2/	file://site A/server A/Disk ID 1/
⋮	⋮	⋮
⋮	⋮	⋮

【図17】

(A)

部分ファイル識別子	所在地	オリジナル所在地
A1	file://site B/server B/Disk ID 3/	file://site A/server A/Disk ID 1/
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
B1	file://site A/server A/Disk ID 1/	file://site A/server A/Disk ID 1/
C1	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮
⋮	⋮	⋮

(B)

部分ファイル識別子	所在地	オリジナル所在地
C1	file://site C/server C/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
A1	file://site B/server B/Disk ID 3/	file://site A/server A/Disk ID 1/
⋮	⋮	⋮
⋮	⋮	⋮

(C)

部分ファイル識別子	所在地	オリジナル所在地
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
C1	file://site C/server C/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮
⋮	⋮	⋮

【図18】

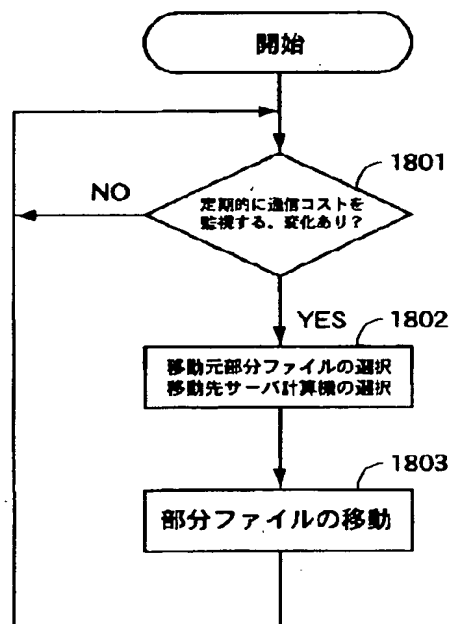


Figure 1 is a block diagram illustrating a system architecture with three sites (A, B, and C) connected to a central network (101).

Site A (1105 サーバ計算機 A):

- 111: 部分ファイル管理部 (Partial File Management Unit)
- 1112: 分散ファイル管理部 (Distributed File Management Unit)
- 1131: 分散ファイル移動部 (Distributed File Movement Unit)
- 113: ネットワークインターフェース (Network Interface)
- 115: 記憶装置 (Storage Device) containing 部分ファイル A1 (Partial File A1) and 部分ファイル B1 (Partial File B1)
- 814: 状態管理部 (Status Management Unit)
- 811: 外部状態管理部 (External Status Management Unit)

Site B (1103 計算機サイト B):

- 132: ネットワークインターフェース (Network Interface)
- 109: クライアント計算機群 B (Client Computer Group B)
- 1106: サーバ計算機 (Server Computer)
- 118: ネットワークインターフェース (Network Interface)
- 1132: 分散ファイル管理部 (Distributed File Management Unit)
- 1117: 分散ファイル移動部 (Distributed File Movement Unit)
- 116: 部分ファイル管理部 (Partial File Management Unit)
- 812: 状態管理部 (Status Management Unit)
- 819: 外部状態管理部 (External Status Management Unit)
- 120: 記憶装置 (Storage Device) containing 部分ファイル C1 (Partial File C1) and 部分ファイル C2 (Partial File C2)

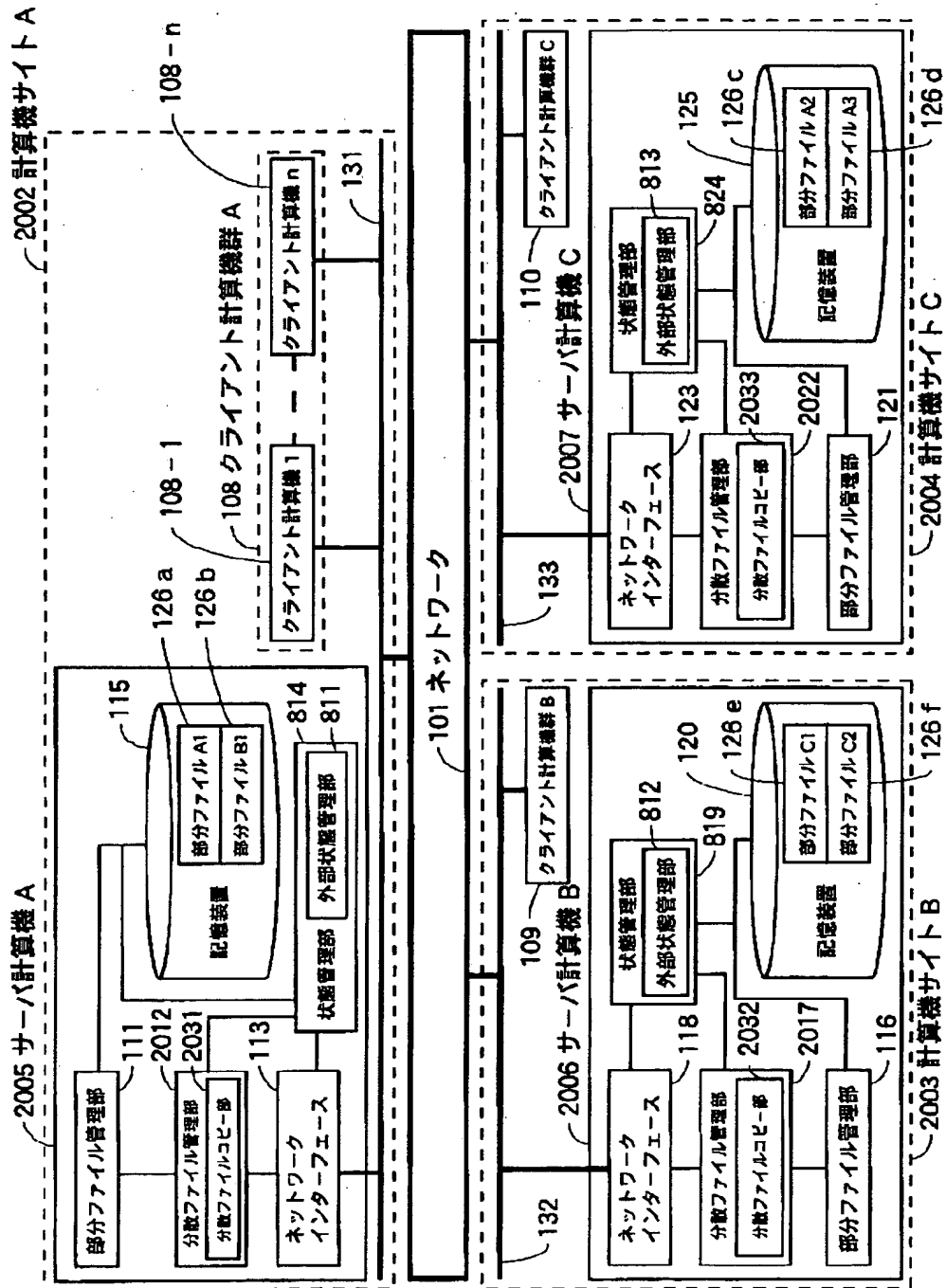
Site C (1104 計算機サイト C):

- 133: ネットワークインターフェース (Network Interface)
- 110: クライアント計算機群 C (Client Computer Group C)
- 1107: サーバ計算機 (Server Computer)
- 123: ネットワークインターフェース (Network Interface)
- 1133: 分散ファイル管理部 (Distributed File Management Unit)
- 1122: 分散ファイル移動部 (Distributed File Movement Unit)
- 121: 部分ファイル管理部 (Partial File Management Unit)
- 813: 状態管理部 (Status Management Unit)
- 824: 外部状態管理部 (External Status Management Unit)
- 125: 記憶装置 (Storage Device) containing 部分ファイル A2 (Partial File A2) and 部分ファイル A3 (Partial File A3)

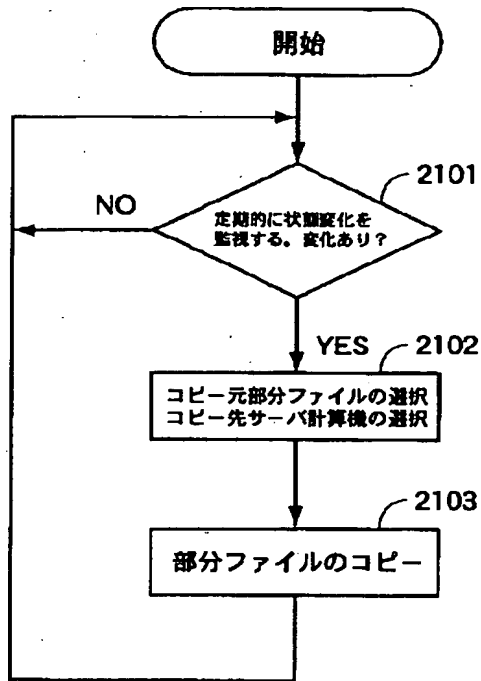
Central Network (101 ネットワーク):

The central network (101) connects all three sites (A, B, and C) and is labeled 101 ネットワーク.

【図20】



【図21】



【図22】

(A)

部分ファイル識別子	所在地	オリジナル所在地
A1	file://site A/server A/Disk ID 1/ file://site C/server C/Disk ID 2/	file://site A/server A/Disk ID 1/
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
B1	file://site A/server A/Disk ID 1/	file://site A/server A/Disk ID 1/
C1	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮
⋮	⋮	⋮

(B)

部分ファイル識別子	所在地	オリジナル所在地
C1	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮
⋮	⋮	⋮

(C)

部分ファイル識別子	所在地	オリジナル所在地
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A1	file://site C/server C/Disk ID 2/	file://site A/server A/Disk ID 1/
⋮	⋮	⋮
⋮	⋮	⋮

【図23】

(A)

部分ファイル識別子	所在地	オリジナル所在地
A1	file://site A/server A/Disk ID 1/ file://site C/server C/Disk ID 2/ file://site B/server B/Disk ID 3/	file://site A/server A/Disk ID 1/
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
B1	file://site A/server A/Disk ID 1/	file://site A/server A/Disk ID 1/
C1	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮
⋮	⋮	⋮

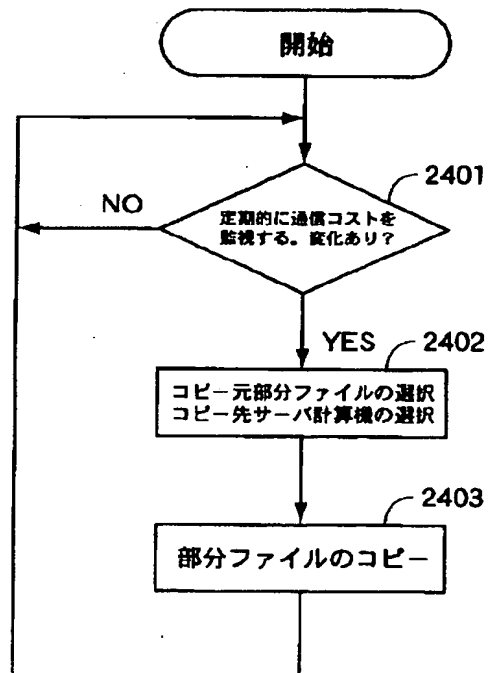
(B)

部分ファイル識別子	所在地	オリジナル所在地
C1	file://site B/server B/Disk ID 3/	file://site B/server B/Disk ID 3/
C2	file://site C/server C/Disk ID 3/	file://site B/server B/Disk ID 3/
A1	file://site B/server B/Disk ID 2/	file://site A/server A/Disk ID 1/
⋮	⋮	⋮
⋮	⋮	⋮

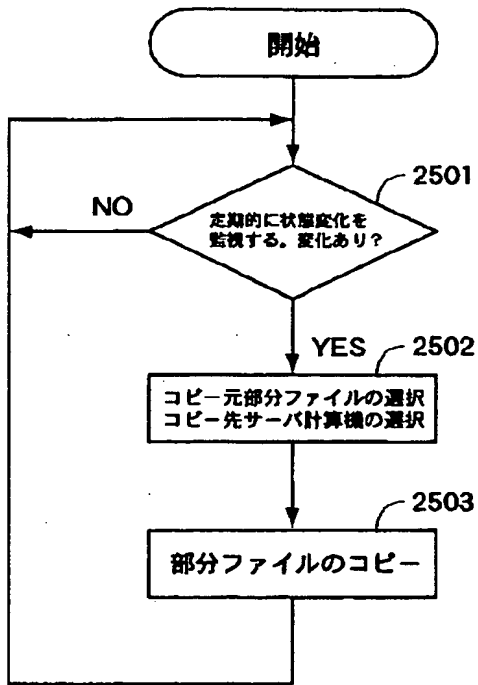
(C)

部分ファイル識別子	所在地	オリジナル所在地
A2	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A3	file://site C/server C/Disk ID 2/	file://site C/server C/Disk ID 2/
A1	file://site C/server C/Disk ID 2/	file://site A/server A/Disk ID 1/
C1	file://site C/server C/Disk ID 3/	file://site B/server B/Disk ID 3/
⋮	⋮	⋮
⋮	⋮	⋮

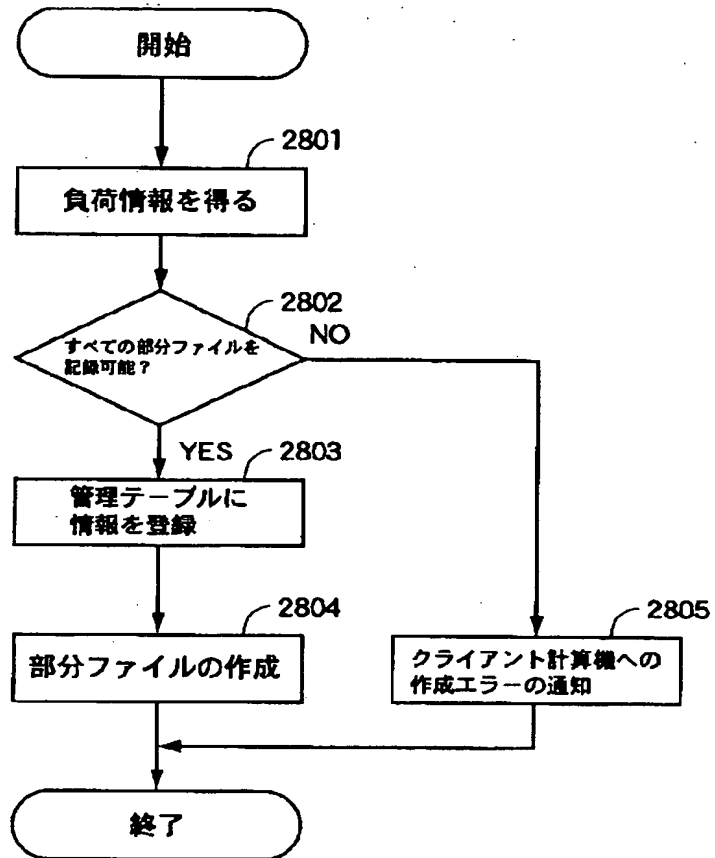
【図24】



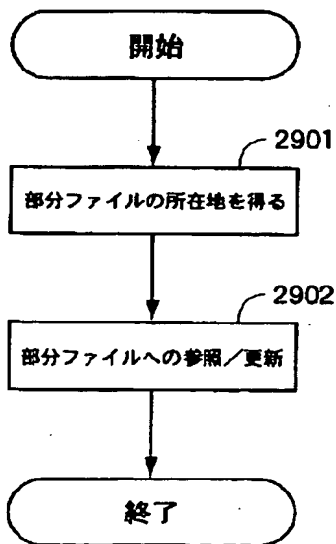
【図25】



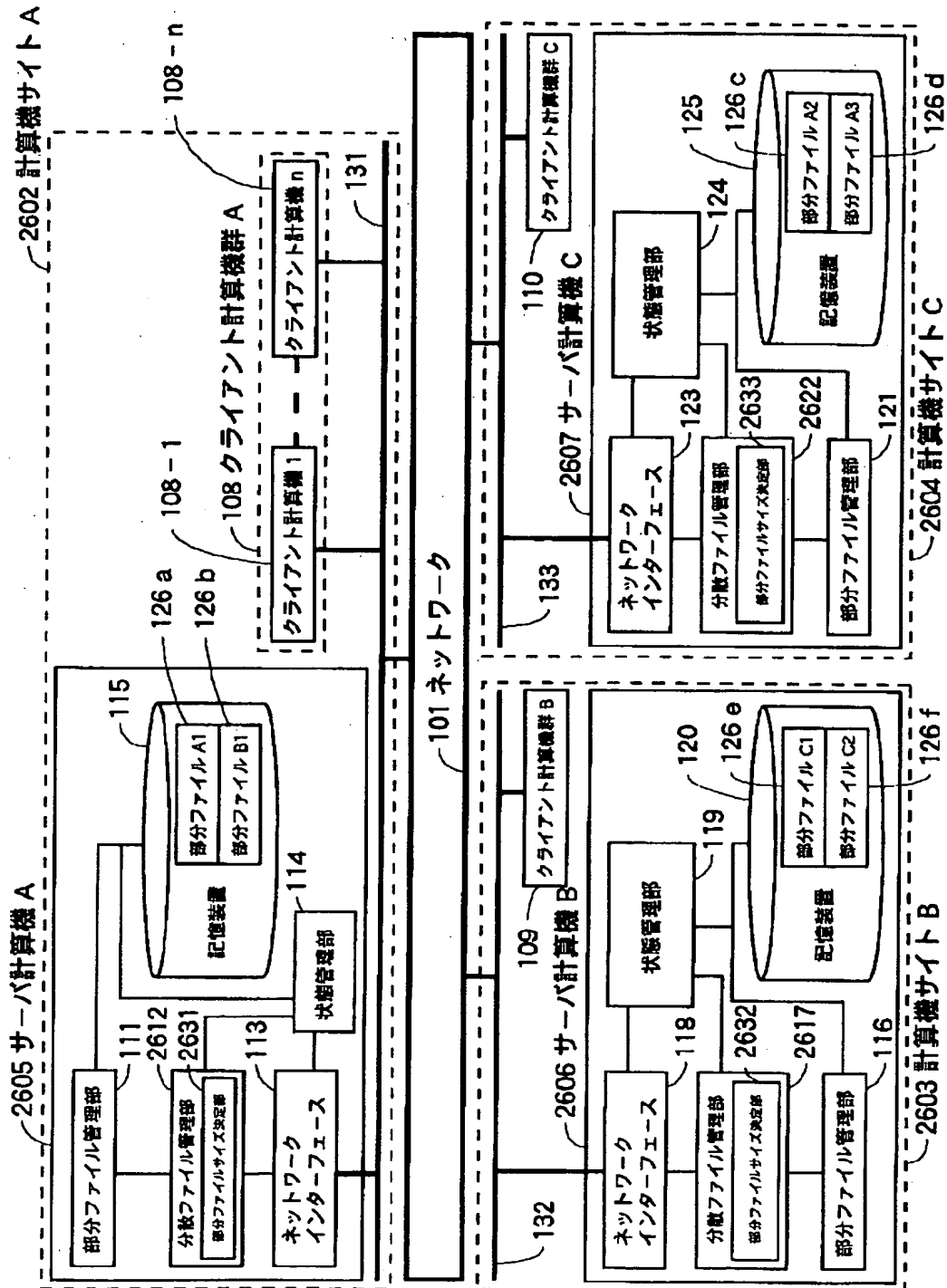
【図28】



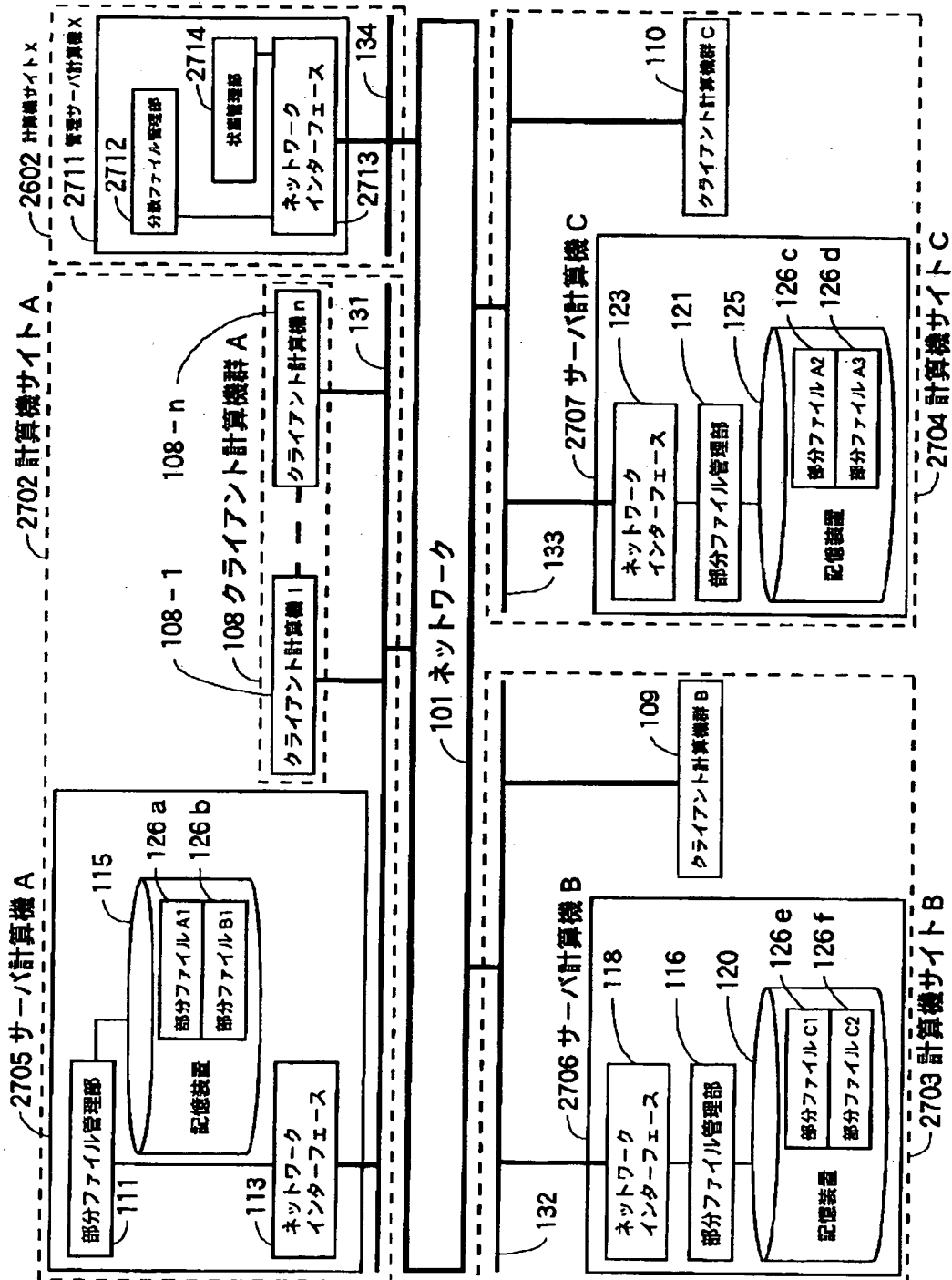
【図29】



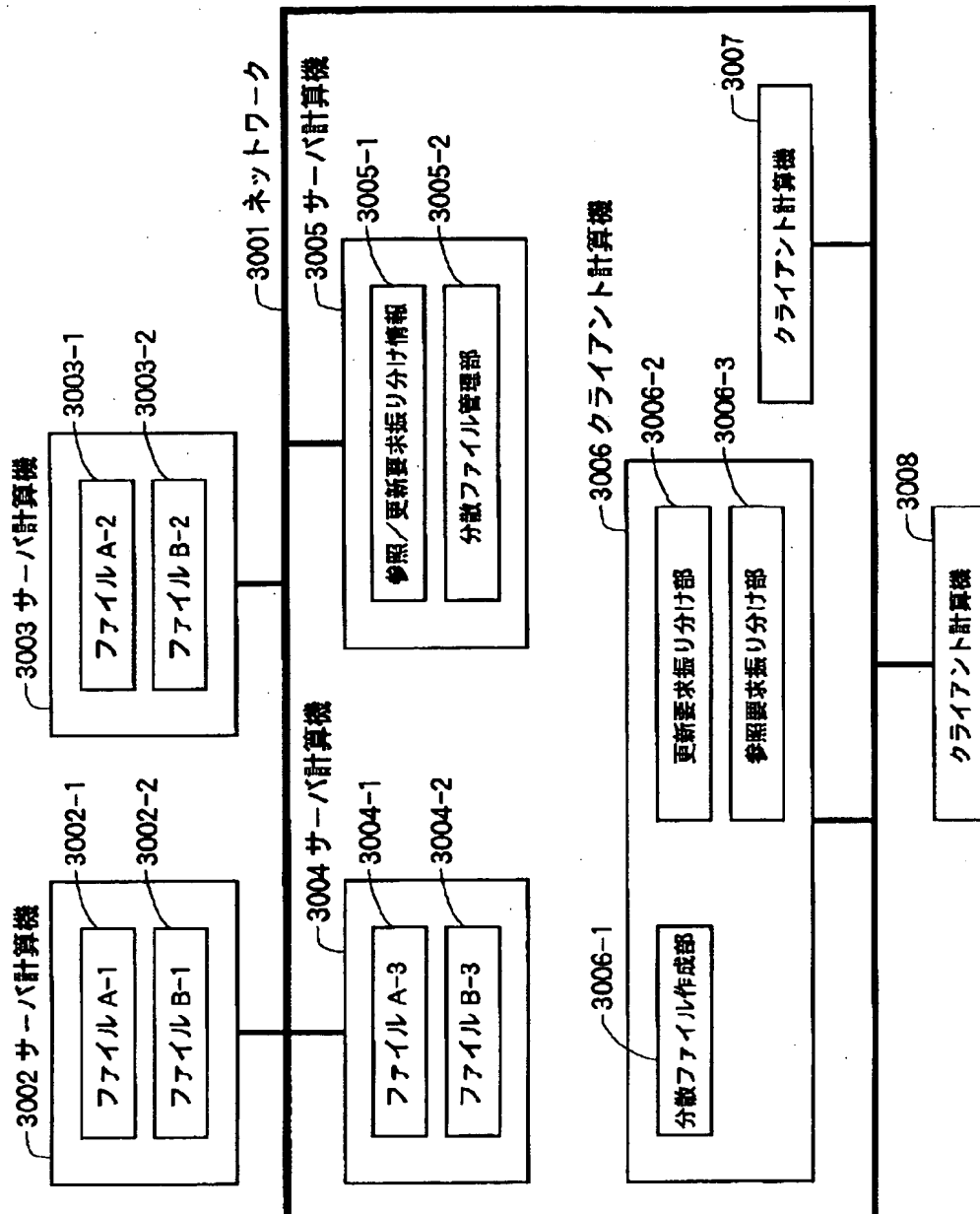
【図26】



【図27】



【図30】



フロントページの続き

(72)発明者 安河内 龍二
大阪府門真市大字門真1006番地 松下電器
産業株式会社内

(72)発明者 田中 則子
大阪府門真市大字門真1006番地 松下電器
産業株式会社内

Fターム(参考) 5B045 BB49 DD16 GG02 GG09 JJ08
5B082 HA01 HA08

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-335144

(43)Date of publication of application : 17.12.1996

(51)Int.Cl.

G06F 3/06

G06F 11/20

G06F 12/16

G06F 13/14

(21)Application number : 07-139781

(71)Applicant : HITACHI LTD

(22)Date of filing : 07.06.1995

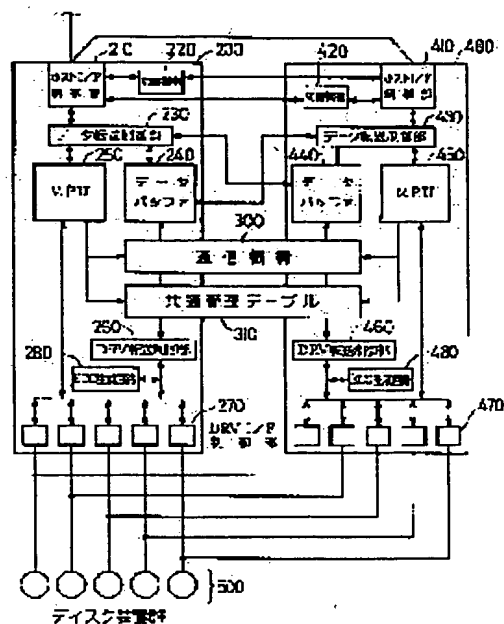
(72)Inventor : MATSUMOTO YOSHIKO
MURAOKA KENJI

(54) EXTERNAL STORAGE DEVICE

(57)Abstract:

PURPOSE: To improve reliability and performance and to provide non-stop maintenance by distributing a load to the plural storage controllers of redundant configuration.

CONSTITUTION: Plural disk drive controllers 200 and 400 of redundant configuration for controlling a disk device are connected to a host device by the same SCSIID, the monitor of mutual operating states and the setting of load distribution information are performed by interposing a communication mechanism 300 and a common managing table 310 and in a normal state, high performance is provided by distributing the load by simultaneously operating the plural disk drive controllers 200 and 400 but in case of fault or maintenance, non-stop operation and non-stop maintenance are provided by executing a switching operation at the degeneracy and recovery caused by disconnection on the side of the fault while using switching mechanism 220 and 420.



LEGAL STATUS

[Date of request for examination]

06.06.2002

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. *** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] External storage containing two or more memory control units which intervene between the storage with which the data delivered and received between the high order equipment characterized by providing the following are stored, and the aforementioned storage and the aforementioned high order equipment, and control transfer of the aforementioned data between the aforementioned high order equipment and the aforementioned storage. An interface means to connect the memory control unit concerned to the aforementioned high order equipment so that two or more aforementioned memory control units may look equivalent [in view of the aforementioned high order equipment]. A surveillance means to be prepared in each aforementioned memory control unit, and to supervise the existence of the obstacle in other aforementioned memory control units, or change instructions. The change means which changes whether which aforementioned memory control unit controls transfer of the aforementioned data between the aforementioned high order equipment by being prepared in each aforementioned memory control unit. A communication-of-information means to transmit the mutual information of the aforementioned memory control unit, and a load-distribution means to make the load which originates in input/output request from the aforementioned high order equipment share among two or more aforementioned memory control units.

[Claim 2] External storage according to claim 1 characterized by providing the following. The data buffer which stores temporarily the aforementioned data which are prepared in each of two or more aforementioned memory control units, and are delivered and received between the aforementioned high order equipment. While writing write request data in alternative or multiplex to each of two or more aforementioned data buffers at the time of the write request from the aforementioned high order equipment. Are at the write-in completion time to the aforementioned data buffer of the aforementioned write request data, and write in to the aforementioned high order equipment and completion is reported. The light after processing in which the aforementioned write request data are made to reflect from the aforementioned data buffer to the aforementioned storage asynchronously [the input/output request from the aforementioned high order equipment]. And they are the data transfer control means which can be performed alternatively about the write-through processing which it is at the write-in completion time to the aforementioned storage of the aforementioned write request data, and writes in to the aforementioned high order equipment and reports completion.

[Claim 3] External storage according to claim 1 or 2 characterized by providing the following. The 1st management information for being made accessible in common from two or more aforementioned memory control units, and discriminating whether each aforementioned memory control unit is healthy. The 2nd management information which specifies any shall be performed between the aforementioned light after processing and the aforementioned write-through processing. A management information storage means by which at least one of the 3rd management information which specifies any of two or more aforementioned memory control units receive the input/output request from the aforementioned high order equipment, and the 4th management information which specifies the assignment of the aforementioned load in each of two or more aforementioned memory control units is stored. The control logic carry out

operation of performing degeneracy operation which continues transfer of the aforementioned data between the aforementioned high order equipment by the aforementioned remaining memory control unit while separating the aforementioned memory control unit which the aforementioned obstacle occurred ignited by generating of an obstacle, or the change instructions from the outside, or was ordered from the outside, and operation return the separated aforementioned memory control unit to a redundant configuration.

[Claim 4] It is the external storage characterized by having the control logic which performs a halt and resumption of alternative write-in operation of the aforementioned write request data to each of the aforementioned data buffer by which the aforementioned data transfer control means were prepared in each of each aforementioned memory control unit in external storage according to claim 2.

[Claim 5] External storage characterized by performing maintenance of the micro program which controls the maintenance or the aforementioned memory control unit of a data buffer corresponding to the stopped aforementioned memory control unit while stopping at least one in two or more aforementioned memory control units alternatively and performing degeneracy operation in external storage according to claim 4.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Industrial Application] Especially this invention is applied to the external-memory subsystem of the redundant configuration which equipped multiplex with the input/output control unit which controls the input/output request of the information from high order equipment etc. about external storage, and relates to effective technology.

[0002]

[Description of the Prior Art] In the external storage which constitutes a computer system, when the memory control unit which intervenes between storage and high order equipment equipped with the storage, and controls transfer of the information between both is not a redundant configuration, if an obstacle occurs in a memory control unit, a subsystem will be obliged to a halt and a rehabilitation work will be performed in the meantime. And an end of this rehabilitation work resumes the business which the memory control unit was rebooted, or the subsystem was rebooted, and had been interrupted till then.

[0003] Moreover, recently, the employment gestalt of operation is increasing 24 hours in various information processing business which uses a computer system, and continuous running is demanded also of the external-memory subsystem. For this reason, for example, the technology whose continuation operation of a system one memory control unit tends to enable by the memory control unit of on stream and an other system stopping, and taking redundant composition about a memory control unit [say / taking a standby state], and changing to the memory control unit of the standby state of an other system at the time of the obstacle of a memory control unit is known as indicated by JP.3-206529A.

[0004]

[Problem(s) to be Solved by the Invention] However, in the above-mentioned conventional technology, although continuous running at the time of an obstacle was possible, in spite of having had two sets of memory control units, actually working is only any one set and it did not change at all with the time of one set efficiently. That is, it was redundant, and also the memory control unit of a system could not but be an object for hot standbys to the last, and could not but be a mere alternative of the memory control unit of an obstacle.

[0005] Moreover, in recent years, the demand to a system was also various, there were various topologies by which an access demand is published from high order equipment to two or more paths to the same storage or separate storage, and it was difficult to build a system in the mere redundant configuration of a memory control unit like before according to various users' request.

[0006] Moreover, in the former, it has the composition that the memory control unit and the data buffer were carried in one board in the cheap system. When performing maintenance control, such as extension of the data buffer in a memory control unit, separation of only a data buffer Eye an impossible hatchet, A data buffer is extended in the state where the system was made to suspend. After the end of extension work, it was impossible to have carried out the maintenance control work of extension etc., having rebooted the memory control unit and the system, having completed the procedure of resuming the business interrupted till then, and processing the

input/output request (I/O) from high order equipment.

[0007] The purpose of this invention is by making two or more memory control units of a redundant configuration distribute a load to offer the external storage which can raise reliability and a performance.

[0008] Other purposes of this invention are to offer the external storage which can realize improvement in the reliability by multiplexing of a memory control unit, and control action with a still more various memory control unit, without making it conscious of the redundant configuration of a memory control unit to a high order equipment side.

[0009] The purpose of further others of this invention is to offer the external storage which can carry out the maintenance control work of the hardware in two or more memory control units of a redundant configuration, software, etc. simple, without stopping operation.

[0010] The purpose of further others of this invention is to offer the external storage which can do the maintenance control work of composition of having carried the memory control unit and the data buffer on the single board during operation.

[0011]

[Means for Solving the Problem] An interface means by which the external storage of this invention connects a memory control unit to high order equipment so that two or more memory control units may look the same [in view of high order equipment]. A surveillance means to supervise other memory control units among two or more memory control units, and the means of communications which can transmit the information between memory control units, It considers as composition including the change means which changes the memory control unit which has received the demand from high order equipment, the input/output request from the high order equipment which one memory control unit received, and the load-distribution means which carries out the load distribution of the processing which accompanies it in two or more memory control units.

[0012] Moreover, the data buffer which once stores the write-in data from high order equipment by preparing for each memory control unit and taking the same redundant configuration as a memory control unit. When the write-in data from high order equipment are stored in a data buffer, while reporting an end to high order equipment and writing the demand with high order equipment in storage from a data buffer asynchronously it considers as composition including a data transfer means to control whether it writes in two or more data buffers of all of a redundant configuration, or it writes in alternatively.

[0013] Moreover, the 1st management information for being made accessible in common from two or more memory control units, and discriminating whether each memory control unit is healthy, The 2nd management information which specifies any shall be performed between light after processing and write-through processing. It considers as composition including a management information storage means by which at least one of the 3rd management information which specifies any of two or more memory control units receive the input/output request from high order equipment, and the management information [of ** the 4th which specifies the assignment of the load in each of two or more memory control units] **s is stored.

[0014]

[Function] In the external storage of this invention, when it is a redundant configuration containing the 1st of a couple, and the 2nd memory control unit, daisy chain connection is made with for example, high order equipment and a SCSI interface, and these the 1st and 2nd memory control units are accessed by the same SCSIID, for example. For example, when the 1st memory control unit has received the input/output request from high order equipment fixed, other 2nd memory control unit distributes based on the 4th management information which specifies the assignment of the load in each of two or more memory control units, and the load accompanying processing of the input/output request concerned can aim at improvement in the throughput of the radial transfer by parallel operation of the improvement in the reliability by the redundant configuration, the 1st, and 2nd memory control units.

[0015] Moreover, he does not need to be [as opposed to / a change / that what is necessary is just to publish an I/O demand to SCSIID also with after / same / high order equipment] an

obstacle by changing the memory control unit which detects an obstacle by the 1st management information and the surveillance means when the 1st memory control unit has received the input/output request from high order equipment fixed, for example and an obstacle occurs in the 1st memory control unit, and receives a demand by the change means to the 2nd memory control unit] conscious at all. Then, the memory control unit acting as the obstacle is separated, and it goes into degeneracy operation. The 1st memory control unit is restored after the maintenance-service end of parts, exchange of a micro program, etc., and it is restored to the original redundant configuration.

[0016] Moreover, have a data buffer in each memory control unit, and the 1st memory control unit receives the write-in data from high order equipment. When light after processing is being performed, a surveillance means detects that the obstacle occurred in the 1st memory control unit, by the change means. When the memory control unit which receives input/output request is changed from the 1st memory control unit to the 2nd memory control unit. Simultaneously, it changes from the multiplex data write-in processing to two or more data buffers to the processing which writes data in the data buffer with which the memory control unit under operation was equipped alternatively.

[0017] At this time, it chooses whether light after processing is performed or write-through processing is performed. This selection is possible when a user sets up the 2nd management information of a management information storage means. That is, when the demand to a user's data reliability is high, and setting it as write-through mode and requiring a performance rather than reliability, it is set as light after mode.

[0018] It changes to the operation written in a data buffer by changing to alternative writing or multiplex writing multiplex after restoration of the 2nd memory control unit, and can restore to a redundant configuration.

[0019] After the 1st memory control unit receives the input/output request from high order equipment, and changing from the processing written in multiplex when double writing is performed to the data buffer of the 1st memory control unit, and the data buffer of the 2nd memory control unit and light after processing is being performed about the write request from high order equipment to the processing written in alternatively, separating the 2nd memory control unit, degenerating it and maintaining extension of a data buffer, exchange of a micro program, etc., the original redundant configuration is restored. Then, the 2nd memory control unit for means of communications of *** which can transmit the information between memory control units notifies completion of maintenance services, such as extension of a data buffer, to the 1st memory control unit, and changes receipt of the demand from high order equipment to self-equipment after a notice using a change means.

[0020] On the other hand, the 1st memory control unit which received the notice degenerates self-equipment, maintains extension of a data buffer etc. and makes it restore. The 1st memory control unit notifies the completion of maintenance of extension of a data buffer etc. to the 2nd memory control unit after restoration using the means of communications which can transmit informational. Ignited by this, it changes from the alternative data writing to a single data buffer to the multiplex write-in processing to two or more data buffers. Thereby, it becomes possible, continuing the radial transfer [business / maintenance control /, such as extension of the data buffer of the 1st memory control unit of a couple, and the 2nd memory control unit, and exchange of a micro program,] between high order equipment.

[0021] Moreover, according to this invention, the 1st memory control unit and the 2nd memory control unit can judge any receive the demand from high order equipment by referring to the 3rd management information which specifies any of two or more aforementioned memory control units set as the management information storage means receive the input/output request from the aforementioned high order equipment. It is also possible for this to receive and process the input/output request in the memory control unit of either the 1st memory control unit or the 2nd memory control unit not only receiving the demand from high order equipment but both. Moreover, it becomes possible by setting up the 3rd management information optionally by the user to specify from a user the memory control unit which receives the demand from high order equipment to be arbitration.

[0022]

[Example] Hereafter, the example of this invention is explained in detail, referring to a drawing.

[0023] Drawing 1 is the conceptual diagram showing an example of the computing system containing the external storage which is one example of this invention. The computing system of this example contains the high order equipment 100 which is a central processing unit, the disk drive control unit 200 and the disk drive control unit 400, and the disk unit 500. The disk drive control unit 200 and the disk drive control unit 400 were connected with high order equipment 100 with the daisy chain of a SCSI interface, the same SCSIID was set up and the disk drive control unit 200,400 has taken the redundant configuration. And in the case of this example, the disk drive control unit 200 receives the demand from high order equipment 100, performs processing which accompanies a demand with the disk drive control unit 200 and the redundant disk drive control unit 400, and controls a disk unit 500. [0024] Drawing 2 is the block diagram showing an example of the internal configuration of the disk drive control units 200 and 400. In addition, since the internal configuration of the disk drive control unit 200 and the disk drive control unit 400 is the same, the disk drive control unit 200 is explained to an example, 2 figures is made the same under the sign of the part which corresponds about the disk drive control unit 400 side, and explanation is omitted.

[0025] a microprocessor unit 250 (Following MPU is called) is performed decoding a RAM (RAM - not shown) serially, and is controlling the whole disk drive control unit 200

[0026] The host I/F control section 210 is performing protocol control with high order equipment 100. The DRV/I/F control section 270 is performing protocol control with each drive. A data buffer 240 is used at the time of the data transfer of the host I/F control section 210 and the DRV/I/F control section 270. Volatilization memory is sufficient as this memory, and non-volatilized memory is sufficient as it. This example describes for an example the case where a data buffer 240 is built by volatilization memory.

[0027] The change mechanism 220 is for changing the host I/F control section which receives I/O from high order equipment 100 to the host I/F control section 210 and the host I/F control section 410 of each disk drive control unit 200 and the disk drive control unit 400. The host I/F control section 210 shall have received in this example. The data transfer control section 230 is controlling the data transfer of high order equipment 100 and a data buffer 240. It has the function of whether this data transfer control section 230 carries out double writing of the light data from high order equipment 100 to the 2nd page, a data buffer 240 and a data buffer 440, or to 1-fold carry out [of only a data buffer 240 / both] writing. Moreover, it is possible to change 1-fold writing or double writing with the directions from MPU250.

[0028] The DRV transfer control section 260 controls the data transfer between a data buffer 240 and a disk unit 500.

[0029] The transmitter style 300 is a mechanism for transmitting the information between MPU250 and MPU450. This transmitter style 300 is enabling the bidirectional transfer between MPU250 and MPU450.

[0030] The common managed table 310 is a managed table in which both reference / renewal of MPU250 and MPU450 are possible.

[0031] this example takes and explains to an example the drive storing method by array composition which is distributed to two or more disk units 500, and stores the logical data from high order equipment 100.

[0032] The ECC generation circuit 280 has the function which generates redundant data to the data sent from high order equipment 100, and can use this function also for the reconstitution of data. 1 logical data unit sent from the high order is sufficient as the unit which adds redundant data, and it is good even to two or more logical data units. This example adds redundant data to four logical data, and describes them in RAID5 method which does not fix the drive which stores this redundant data.

[0033] Next, with reference to drawing 3, an example of the composition of the common managed table 310 is explained. It is used for surveillance intelligence 320 confirming whether each disk drive control units 200/400 are operating normally. Surveillance intelligence A321 sets up information at a fixed interval, when MPU250 of the disk drive control unit 200 is normally

judged that operation is possible. Moreover, when MPU250 judges normally that operation is impossible, the information which shows abnormalities is set up. In addition, MPU450 of the disk drive control unit 400 as well as MPU250 sets information as surveillance intelligence B322. [0034] The data transfer mode information 330 directs the end report opportunity to the light data write request from high order equipment 100 at the time of the degenerate state of a system. That is, it is this information at the write-in completion time to a data buffer 240 or a data buffer 440, and is the information for judging whether an end is reported to high order equipment 100, or an end report is carried out when it writes even in a disk unit 500 from a data buffer 240 (write-through mode being called below) (light after mode being called below). [0035] The directions information on a disk drive control unit that the host I/O receipt information 340 receives I/O between two disk drive control units 200/400 is shown. This example explains as that by which the disk drive control unit 200 is set as the host I/O receiving side.

[0036] The load-distribution information 350 is information for carrying out the load distribution of the processing accompanying I/O from high order equipment between [of two] disk drive control unit 200 / 400. The method of a load distribution may divide into each disk drive control unit the disk unit made applicable to access, and may share it with the processing which stores light data in a disk unit 500 from a data buffer with asynchronous processing of the I/O demand from high order equipment 100 and I/O demand from high order equipment 100. Or the method of performing processing is sufficient as the way which writes in all the matters that must be processed into load-distribution information, considers as competition logic between two MPU, and has an opening as MPU.

[0037] By this example, processing of the I/O demand from high order equipment 100 and the I/O demand from high order equipment 100 explain the method shared with the processing which stores light data in a disk unit 500 from data buffers 240/440 asynchronously. Therefore, in this example, the information on the light data stored in data buffers 240/440 shall be contained in the load-distribution information 350.

[0038] Next, the write-in processing and reading processing of data to a disk unit 500 are explained from the high order equipment 100 in the computing system in this example. [0039] Usually, at the time of the write request from high order equipment 100, by the host I/F control section 210, the disk drive control unit 200 receives write-in logical data, stores it in a data buffer 240 and a data buffer 440 doubly by the data transfer control section 230, sets storing information as the load-distribution information 350 on the common managed table 310, and reports an end to high order equipment 100 at this time. Serially, if MPU450 has storing information with reference to the load-distribution information 350. The light data concerned and the data of the same address already stored in the drive (the old data are called below). The parity data corresponding to the light data concerned are read from a disk unit 500 by the DRV/F control section 470 and the DRV transfer control section 460. Light data, the old data, and parity data generate the parity data (new parity data are called below) corresponding to light data in the ECC generation circuit 480. Light data are stored in a disk unit 500 by writing the new parity data and light data which were generated in a disk unit 500 by the DRV/F control section 470 and the DRV transfer control section 460. This processing is asynchronously performed with the I/O demand from high order equipment 100. Moreover, the read-out processing of the old data / old parity data performed since light data are stored and new parity generation processing, and new parity data storage processing are called light penalty in RAID5. [0040] Thus, the storing demand of the light data from high order equipment 100 is processing that a load is very high, when operating two or more disk units 500 as disk array equipment. Efficiency leads to the improvement in a performance as a system well rather than carrying out a role assignment and performing this processing with two disk drive control units 200,400 performs by one set only of a disk drive control unit. A cheap processor is carried especially as latest commercial-scene trend, and it has become a very important element with high performance and high-reliability to reduce system-wide cost. Therefore, in light penalty processing, although that much accesses to a drive occur also leads to performance degradation, since the transit time of the micro program of the processor which controls it before it is long, a

processor neck has many bird clappers as a system. At this time, the performance near the double precision can be taken out with processing by two sets of the disk drive control units 200 and 400 like this example.

[0041] Next, the reading demand from high order equipment 100 and MPU250 start reading of data from a physical drive (disk unit 500) by the DRV/F control section 270 and the DRV transfer control section 260, and transmit it to high order equipment 100. Moreover, while the lead demand address from high order equipment 100 is continuing at this time, it may judge that the disk drive control unit 400 is sequential lead processing, and processing which reads asynchronously [I/O from high order equipment 100] an certain amount of data which follows the lead demand address from high order equipment 100 to data buffers 240 and 440 may be performed. When there is next an I/O demand from high order equipment by carrying out like this, the target data are already stored in data buffers 240/440, and data can be transmitted without producing access to the disk unit 500 which time requires, and it leads to the improvement in a performance as the whole.

[0042] As mentioned above, though it is a redundant configuration, it leads not only to reliability but to improvement in a performance by performing a part of processing rather than making a redundant portion (this example disk drive control unit 400) only stand by as an object for the change at the time of obstacle generating.

[0043] Next, in this example, while two sets of the disk drive control units 200/400 perform processing, operation which performs automatic switching and restoration at the time of an obstacle is explained. First, the surveillance procedure which detects an obstacle automatically is explained.

[0044] MPU 250 and 450 sets the information (normal information is called hereafter) which shows that MPU450 of MPU250 is normal to surveillance intelligence 321 whenever fixed time passes controlling the disk drive control units 200 and 400 as surveillance intelligence 322. However, in order to show having set up for every fixed time, the information which changes serially is set to this information. For example, it is the information which is added one [at a time]. Moreover, when accessing a data buffer is that each MPU250,450 judged normally that operation was impossible with the disk drive control unit 200,400 concerned impossible from MPU, for example, the information (this is called obstacle information below) which shows that it is an obstacle is set as surveillance intelligence. Hereafter, the flow chart of drawing 4 explains an example of the above-mentioned surveillance procedure.

[0045] Here, MPU450 of the disk drive control unit 400 takes and explains to an example operation which supervises the disk drive control unit 200 of an other system.

[0046] MPU250 judges first whether fixed time passed at Step 600. If fixed time has not passed, it progresses to Step 608 and it is judged that the disk drive control unit 200 is normal.

[0047] If fixed time has passed, it will progress to Step 601 and the normal information which shows that MPU450 is normal will be set up. And it progresses to Step 602 and the surveillance intelligence 322 of the disk drive control unit 200 is referred to. If it judges whether this information is normal and judges that it is normal at Step 603, it will progress to Step 604. If it judges that it is an obstacle, it will progress to Step 607 and it will be judged that the disk drive control unit 200 is an obstacle.

[0048] At the time of normal information, it progresses to Step 605 and judges at Step 605 whether this normal information had change from before. That is, MPU250 may have fallen impossible [a setup of surveillance intelligence] according to the obstacle of a micro program etc. Such an obstacle is judged with the check of this step 605. If there is change, it will progress to Step 608 and it will be judged that it is normal. When there is no change, it progresses to Step 606 and judges whether the time of a margin longer than fixed time has passed. Consequently, if it has passed, it progresses to Step 607 and is judged as an obstacle, and if it has not passed, it will progress to Step 608 and it will be judged that it is normal. According to the above surveillance procedure, both of obstacles of a micro program can also detect the obstacle of hardware simultaneously.

[0049] Next, an example of the processing from which the disk drive control unit 400 recognizes the obstacle of the disk drive control unit 200 of an other system, and changes with reference to

the flow chart of drawing 5 is explained.

[0050] Refer to the load-distribution information 350 for MPU450 serially at Step 700 first. Consequently, if the light data from high order equipment 100 do not exist in a data buffer 240 and 440 at Step 701, it progresses to Step 704. If it exists, in order to progress to Step 702 and to generate the parity corresponding to the light data of a data buffer 440, the old data and the old parity data corresponding to the light data concerned are read from a disk unit 500, and new parity data are generated in the ECC generation circuit 480. Then, it progresses to Step 703 and light data and new parity data are stored in a disk unit 500 by the DRV transfer control section 460 and the DRV/I/F control section 470. Next, at Step 704, the obstacle of the disk drive control unit 200 is checked in the surveillance procedure after Step 600 of drawing 4.

Consequently, if normal, it will progress to Step 700 and processing will be continued. If it judges that a change is required, it will progress to Step 710 and reception of I/O from high order equipment 100 will be changed from the disk drive control unit 200 to the disk drive control unit 400 using change procedure. And the disk drive control unit 400 substitutes Step 720 for the I/O processing from high order equipment 100 which the disk drive control unit 200 was performing, and it is performed.

[0051] Next, it changes with the flow chart of drawing 6, and an example of procedure is explained.

[0052] At Step 711, it directs to write the data concerned in a data buffer 440 one-fold to the data transfer control section 430 at the time of the light data receipt from high order equipment 100 first. That is, since a data buffer does not exist in the disk drive control unit 400 until it exchanges the part which the disk drive control unit 200 was degenerated, separated, and carried out obstacle generating and restores, since the obstacle occurred in the disk drive control unit 200, double writing like [at the time of a normal redundant configuration] is not made.

[0053] And it directs to change the I/O demand from high order equipment 100 from the host I/F control section 210 to the host I/F control section 410 by the change mechanism 420 by Step 712. Although it stops receiving the demand from high order equipment 100, as for the host I/F control section 210, the host I/F control section 410 will come to receive the demand from high order equipment 100 and a disk drive control unit will change substantially by this, at this example, there is no need of knowing the disk drive control unit by the side of receipt having changed that SCSIID should just publish eye the same hatchet and high order equipment 100 to SCSIID same before changing I/O.

[0054] Next, after changing using the flow chart of drawing 7, an example of a procedure which performs I/O with the disk drive control unit 400 is explained.

[0055] If I/O processing is received from high order equipment 100 at Step 721, it will progress to Step 722 and a lead demand or a light demand will be judged. At the time of a lead demand, it progresses to Step 729 and object data are read into a data buffer 440 from the disk unit 500 corresponding to the lead demand concerned. It progresses to Step 730, data are transmitted to high order equipment 100 from a data buffer 440, and Step 728 reports an end to high order equipment 100.

[0056] At the time of a light demand, it progresses to Step 723 and light data are stored in a data buffer 440. Furthermore, it progresses to Step 724 and judges whether it is write-through mode at Step 725 with reference to the data transfer mode information 330. Consequently, it progresses to Step 728 at the time of the mode in which an end is reported to high order equipment 100 when stored at the time 440 of light after mode, i.e., a data buffer, it reports an end, and stores it in a disk unit 500 from a data buffer 440 asynchronously after that. It progresses to Step 726 at the time of write-through mode, it creates the parity data to light data, stores light data and new parity data in a disk unit 500 at Step 727, and reports an end at Step 728. Furthermore, after this, Step 703 is performed from Step 700 in the flow chart of drawing 5, and processing before a change is also performed.

[0057] Thus, according to this example, mutual change operation and continuation of processing are automatically possible without the directions from high order equipment 100 at the disk drive control units 200/400, without carrying out no consciousness to high order equipment 100.

[0058] Next, the disk drive control unit 200 is restored and an example of the method when returning to the original redundant configuration is explained.

[0059] First, the flow chart shown in drawing 8 explains an example of restoration operation by the side of the disk drive control unit 200. It notifies that restoration was completed to the disk drive control unit 400 by Step 810 at transmitter guard 300. Then, the disk drive control unit 200 turns into a redundant disk drive control unit, and before and a position interchange. At Step 811, asynchronous DESUTEJ processing (Steps 700-705 of drawing 5) which the disk drive control unit 400 was performing is performed before.

[0060] Furthermore, an example of operation of the near disk drive control unit 400 which received the notice with the flow chart shown in drawing 9 is explained.

[0061] If the completion of restoration of the disk drive control unit 200 is recognized at transmitter guard 300 by Step 820, it will point to double writing to data buffers 240 and 440 to the data transfer control section 430 at Step 821, and only I/O processing from high order equipment 100 will be performed at Step 821. Thus, receiving the I/O demand from high order equipment 100, restoring in the original redundant composition is possible, and improvement in a performance can also be aimed at by carrying out the load distribution of the processing with two more disk drive control units 200/400.

[0062] Next, it explains, referring to the flow chart of drawing 10 about an example of the extension method of the data buffer under operation of the disk drive control units 200/400. In addition, let the disk drive control unit which has received I/O from high order equipment 100 be the disk drive control unit 200.

[0063] When there is an extension demand of a data buffer, the disk drive control unit concerned judges whether it is an I/O receiving side at Step 911. The content of processing of the disk drive control unit 200 is explained first. Since the disk drive control unit 200 is an I/O receiving side, it progresses to Step 912, the disk drive control unit 400 degenerates first, and it recognizes separating. Then, it directs to make light data into 1-fold writing to a data buffer 240 to the data transfer control section 230. Then, I/O processing from high order equipment 100 is performed at Step 913, and Step 700 of drawing 5 - Step 703 are performed at Step 914. That is, it substitutes for a part to have performed with the disk drive control unit 400. It waits for the completion of restoration of the disk drive control unit 400, repeating Step 913 and Step 914.

[0064] Next, as for the disk drive control unit concerned, the disk drive control unit 400 also judges whether it is an I/O receiving side at Step 911. Since it is not an I/O receiving side as a result, it progresses to Step 915, and the disk drive control unit 400 concerned detaches, and extends a data buffer 440 at Step 916. It notifies having restored at Step 917 to the disk drive control unit 200 through the transmitter style 300 after the completion of extension.

[0065] Since the disk drive control unit 200 needs to extend shortly, the disk drive control unit 400 changes the host I/F control section which carries out I/O reception using the change mechanism 420 at Step 919 to self-# in order to substitute for receipt of I/O. Then, I/O processing from high order equipment 100 is performed at Step 920, Step 700 of drawing 5 - Step 703 are performed at Step 921, and it waits for the completion of restoration of the disk drive control unit 200.

[0066] The disk drive control unit 200 which has recognized restoration through the transmitter style 300 at Step 918 is separated at Step 922, and extends a data buffer 240 at Step 923. The transmitter style 300 notifies restoration to the disk drive control unit 400 at Step 924 after the completion of extension. After a notice, since the disk drive control unit 200 concerned is not a host I/O receiving side, it turns to the side which performs Step 700 of drawing 5 - Step 705 at Step 925.

[0067] The disk drive control unit 400 will direct to write light data to data buffers 240/440 doubly to the data transfer control section 230 at Step 927, if restoration of an other system is recognized at transmitter guard 300 by Step 926. I/O processing from high order equipment 100 is performed at Step 928.

[0068] Thus, though I/O from high order equipment 100 is performed, extension of the data buffers 240/440 of each # is attained. That is, according to this example, by the former, extension of extension of a data buffer is attained in online to having been unrealizable unless it

was after suspending a system. Especially when the disk drive control unit was built on one board realized in the low cost, the exchange for every board was impossible for extension under eye a required hatchet and operation. In this example, extension of a data buffer is possible, setting to the disk drive control units 200/400 of a redundant configuration, and degenerating / restoring one set at a time.

[0069] Moreover, according to this example, by transposing processing of Step 916 of drawing 10, and Step 923 to micro program exchange work, exchange of the micro program under operation is possible, and the demand of 24-hour operation is effective in especially the maintenance control work in a remarkable computer system in recent years.

[0070] Moreover, a user can direct whether for the light demand from high order equipment 100 to be written in by the data buffer, and to report an end, or to write even in a disk unit 500 and report an end during the degeneracy at the time of a piece system obstacle. That is, you may perform automatically rewriting of this data transfer mode information 330 by a user's program. That is, if an end is reported when a data buffer becomes 1st page composition, and stored in data FABBA, although it excels in responsibility, a data guarantee becomes impossible when an obstacle occurs in a disk drive control unit at this time. Since light penalty processing occurs in on the other hand storing even in a disk unit 500, although responsibility will deteriorate considerably, a positive response can be reported to high order equipment 100, and it is reliable. In the case of the external storage of this example, according to the demand level of the reliability to the file which a user treats, it can choose optionally whether priority is given to reliability, or priority is given to a speed of response with directions of a user, and it becomes possible to build a flexible file system.

[0071] Furthermore by this invention, two or more disk drive control units can also offer simultaneously the system which can be accessed not only from redundant composition but from two or more high order equipment or two or more buses. This example of a system configuration is shown in drawing 11 and drawing 12.

[0072] Although drawing 11 is the same composition as drawing 1 of an example explained until now, when I/F with high order equipment 100 is SCSI, with the composition of drawing 1, the points connected by SCSIID from which a memory control unit 0 (400A) and a memory control unit 1 (200A) differ by the composition of drawing 11 to the memory control unit 0 and the memory control unit 1 having been connected by the same SCSIID. In the composition of this drawing 11, both receive and process an I/O demand from high order equipment 100. Moreover, drawing 12 is the block diagram showing an example of the system configuration to which two or more memory control units 0 (400B) and memory control units 1 (200B) were connected by the multi-pass to the same high order equipment 100. The composition of this drawing 12 of a memory control unit 0 (400B) and a memory control unit 1 (200B) is all also an execute permission about the I/O demand from high order equipment 100. Specification of any perform an I/O demand is realized by rewriting the host I/O receipt information 340 of the common managed table 310. That is, each memory control unit determines first whether the memory control unit concerned receives I/O from high order equipment with reference to the host I/O receipt information 340. Thus, in this invention, it can respond to various users' connection method, and a flexible system can be built.

[0073] As explained above, while two or more disk drive control units 200/400 of a redundant configuration carry out a load distribution, according to this example, the file system which can realize simultaneously not only the improvement in reliability but improvement in a performance can be offered by performing the demand from high order equipment 100. Moreover, performing the I/O demand from high order equipment 100, while all the disk drive control units 200/400 carry out a load distribution, but, it changes automatically, operation is continued, without looking for directions in any way from high order equipment 100 at the time of obstacle generating, and it becomes possible to restore further. It becomes exchangeable [extension of a data buffer, or a micro program] by this, performing the I/O demand from high order equipment 100, and non-stopped maintenance can be realized. Moreover, not only a redundant configuration but all disk drive control units are possible also for making it the composition which receives the demand from high order equipment 100 simultaneously, and can respond to the various file systems

which a user demands flexibly.

[0074]

[Effect of the Invention] According to the external storage of this invention, the effect that reliability and a performance can be raised is acquired by making two or more memory control units of a redundant configuration distribute a load.

[0075] Moreover, the effect that improvement in the reliability by multiplexing of a memory control unit and control action with a still more various memory control unit are realizable is acquired, without making it conscious of the redundant configuration of a memory control unit to a high order equipment side.

[0076] Moreover, the effect that the maintenance control work of the hardware in two or more memory control units of a redundant configuration, software, etc. is executable simple is acquired, without stopping operation.

[0077] Moreover, the effect that the maintenance control work of composition of having carried the memory control unit and the data buffer on the single board can be done during operation is acquired.

[Translation done.]

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.

2.*** shows the word which can not be translated.

3. In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the conceptual diagram showing an example of the computing system containing the external storage which is one example of this invention.

[Drawing 2] It is the block diagram showing an example of the internal configuration of the disk drive control unit which constitutes the external storage which is one example of this invention.

[Drawing 3] It is the conceptual diagram showing an example of the composition of the common managed table used in the external storage which is one example of this invention.

[Drawing 4] It is the flow chart which shows an example of an operation of the external storage which is one example of this invention.

[Drawing 5] It is the flow chart which shows an example of an operation of the external storage which is one example of this invention.

[Drawing 6] It is the flow chart which shows an example of an operation of the external storage which is one example of this invention.

[Drawing 7] It is the flow chart which shows an example of an operation of the external storage which is one example of this invention.

[Drawing 8] It is the flow chart which shows an example of an operation of the external storage which is one example of this invention.

[Drawing 9] It is the flow chart which shows an example of an operation of the external storage which is one example of this invention.

[Drawing 10] It is the flow chart which shows an example of an operation of the external storage which is one example of this invention.

[Drawing 11] It is the conceptual diagram showing the modification of a topology with the high order equipment in the external storage which is one example of this invention.

[Drawing 12] It is the conceptual diagram showing the modification of a topology with the high order equipment in the external storage which is one example of this invention.

[Description of Notations]

100 [-- Host I/F control section,] -- High order equipment, 200 -- A disk drive control unit, 210 220 [-- Data buffer,] -- A change mechanism, 230 -- A data transfer control section, 240 250 -- A microprocessor unit, 260 -- DRV transfer control section, 270 [-- Transmitter style,] -- An DRV/I/F control section, 280 -- An ECC generation circuit, 300 310 [-- Surveillance intelligence,] -- A common managed table, 320 -- Surveillance intelligence, 321 322 [-- Host I/O receipt information,] -- Surveillance intelligence, 330 -- Data transfer mode information, 340 350 [-- Host I/F control section,] -- Load-distribution information, 400 -- A disk drive control unit, 410 420 [-- A data buffer, 450 / -- A microprocessor unit, 460 / -- A DRV transfer control section, 470 / -- An DRV/I/F control section, 480 / -- An ECC generation circuit, 500 / -- Disk unit,] -- A change mechanism, 430 -- A data transfer control section, 440

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-335144

(43) 公開日 平成8年(1996)12月17日

(51) Int.Cl. ⁸	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 3/06	3 0 4		G 0 6 F 3/06	3 0 4 B
11/20	3 1 0		11/20	3 1 0 B
12/16	3 1 0	7623-5B	12/16	3 1 0 Q
13/14	3 1 0	7368-5E	13/14	3 1 0 F

審査請求 未請求 請求項の数 5 O L (全 17 頁)

(21) 出願番号 特願平7-139781

(22) 出願日 平成7年(1995)6月7日

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 松本 佳子

神奈川県小田原市国府津2880番地 株式会

社日立製作所ストレージシステム事業部内

(72) 発明者 村岡 健司

神奈川県小田原市国府津2880番地 株式会

社日立製作所ストレージシステム事業部内

(74) 代理人 弁理士 筒井 大和

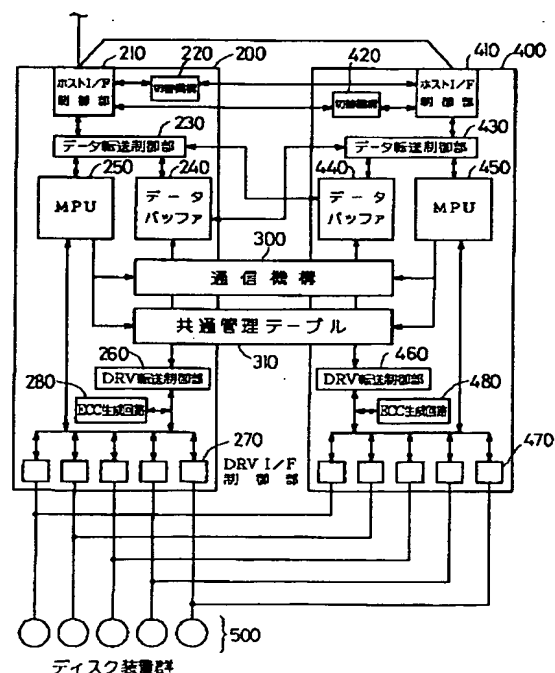
(54) 【発明の名称】 外部記憶装置

(57) 【要約】

【目的】 冗長構成の複数の記憶制御装置に負荷を分散させて、信頼性および性能を向上させるとともに、無停止保守を実現する。

【構成】 ディスク装置を制御する冗長構成の複数のディスクドライブ制御装置200および400を上位装置に対して同一SCSI IDで接続するとともに、通信機構300および共通管理テーブル310を介在させて相互の稼働状態の監視および負荷分散情報の設定を行い、正常時には、複数のディスクドライブ制御装置200および400の同時稼働による負荷分散によって高性能を実現し、障害や保守作業時には、切替機構220および420によって障害側の切り離しによる縮退および復旧時の切り替え操作を実行して、無停止稼働および無停止保守を実現する。

図2



【特許請求の範囲】

【請求項1】 上位装置との間で授受されるデータが格納される記憶装置と、前記記憶装置と前記上位装置との間に介在し、前記上位装置と前記記憶装置との間における前記データの授受を制御する複数の記憶制御装置とを含む外部記憶装置であって、複数の前記記憶制御装置が前記上位装置からみて等価に見えるように当該記憶制御装置を前記上位装置に接続するインターフェイス手段と、個々の前記記憶制御装置に設けられ、他の前記記憶制御装置における障害または切替指令の有無を監視する監視手段と、個々の前記記憶制御装置に設けられ、いずれの前記記憶制御装置が前記上位装置との間における前記データの授受の制御を行うかを切り替える切替手段と、前記記憶制御装置の相互間における情報の伝達を行う情報伝達手段と、前記上位装置からの入出力要求に起因する負荷を複数の前記記憶制御装置間にて分担させる負荷分散手段と、を備えたことを特徴とする外部記憶装置。

【請求項2】 請求項1記載の外部記憶装置において、複数の前記記憶制御装置の各々に設けられ、前記上位装置との間で授受される前記データを一時的に格納するデータバッファと、前記上位装置からの書き込み要求時、複数の前記データバッファの各々に対して書き込み要求データを選択的または多重に書き込むとともに、前記書き込み要求データの前記データバッファに対する書き込み完了時点で前記上位装置に対して書き込み完了を報告し、前記上位装置からの入出力要求とは非同期に前記データバッファから前記記憶装置へ前記書き込み要求データを反映させるライトアフト処理、および前記書き込み要求データの前記記憶装置に対する書き込み完了時点で前記上位装置に対して書き込み完了を報告するライトスルー処理を選択的に実行可能なデータ転送制御手段と、を備えたことを特徴とする外部記憶装置。

【請求項3】 請求項1または2記載の外部記憶装置において、複数の前記記憶制御装置から共通にアクセス可能にされ、個々の前記記憶制御装置が健全か否かを識別するための第1の管理情報、前記ライトアフト処理および前記ライトスルー処理の何れを実行するかを指定する第2の管理情報、複数の前記記憶制御装置の何れが前記上位装置からの入出力要求を受け付けるかを指定する第3の管理情報、複数の前記記憶制御装置の各々における前記負荷の分担を指定する第4の管理情報の少なくとも一つが格納される管理情報記憶手段と、障害の発生または外部からの切替指令を契機として、前記障害が発生したか、または外部から指令された前記記憶制御装置を切り離すとともに、残りの前記記憶制御装

置によって前記上位装置との間における前記データの授受を継続する縮退運転を行う操作、および切り離されていた前記記憶制御装置を冗長構成に復帰させる操作を行う制御論理と、を備えたことを特徴とする外部記憶装置。

【請求項4】 請求項2記載の外部記憶装置において、前記データ転送制御手段は、個々の前記記憶制御装置の各々に設けられた前記データバッファの各々に対する前記書き込み要求データの選択的な書き込み操作の停止および再開を行う制御論理を備えたことを特徴とする外部記憶装置。

【請求項5】 請求項4記載の外部記憶装置において、複数の前記記憶制御装置の中の少なくとも一つを選択的に停止させて縮退運転を行うとともに、停止された前記記憶制御装置に対応するデータバッファの保守または前記記憶制御装置を制御するマイクロプログラムの保守を実行することを特徴とする外部記憶装置。

【発明の詳細な説明】**【0001】**

【産業上の利用分野】 本発明は、外部記憶装置に関し、特に、上位装置からの情報の入出力要求を制御する入出力制御装置を多重に備えた冗長構成の外部記憶サブシステム等に適用して有効な技術に関する。

【0002】

【従来の技術】 コンピュータシステムを構成する外部記憶装置においては、記憶媒体を備えた記憶装置と上位装置との間に介在して両者間の情報の授受を制御する記憶制御装置が冗長構成でない場合、記憶制御装置に障害が発生するとサブシステムは停止を余儀なくされ、この間に復旧作業がおこなわれる。そして、この復旧作業が終了すると、記憶制御装置が再起動され、あるいは、サブシステムが再起動されそれまで中断していた業務が再開される。

【0003】 また最近では、コンピュータシステムを用いる様々な情報処理業務において24時間稼働の運用形態が増加しており、外部記憶サブシステムにも連続運転が要求されている。このため、たとえば、特開平3-206529号公報に記載されているように、一方の記憶制御装置が運転中、他系の記憶制御装置は停止してスタンバイ状態をとるといふ、記憶制御装置に関して冗長な構成をとり、記憶制御装置の障害時、他系のスタンバイ状態の記憶制御装置に切り替わることにより、システムを継続運転可能にしようとする技術が知られている。

【0004】

【発明が解決しようとする課題】 しかし、上述の従来技術においては、障害時の連続運転は可能であるが、2台の記憶制御装置を備えているにもかかわらず、実際に稼働するのはいずれか1台のみであり性能的には1台の時となら変わらなかった。すなわち冗長な他系の記憶制御装置はあくまでホットスタンバイ用であり、障害の記

憶制御装置の単なる代替でしかなかった。

【0005】また、近年では、システムへの要求も様々であり、上位装置からも複数の経路より、同一の記憶装置へ、または別々の記憶装置へとアクセス要求が発行されるような様々な接続形態があり、従来のような記憶制御装置の単なる冗長構成では、多様なユーザの要請に合わせてシステムを構築することが困難であった。

【0006】また、従来において、安価なシステムでは1つのボード内に記憶制御装置とデータバッファが搭載された構成となっており、記憶制御装置内のデータバッファの増設等の保守管理を行う場合、データバッファのみの切り放しが不可能なため、システムを一旦停止させた状態でデータバッファを増設し、増設作業の終了後、記憶制御装置やシステムを再起動させ、それまで中断していた業務を再開する、という手順を踏む必要があり、上位装置からの入出力要求（I/O）を処理しながら増設等の保守管理作業を遂行することは不可能であった。

【0007】本発明の目的は、冗長構成の複数の記憶制御装置に負荷を分散させることにより、信頼性および性能を向上させることが可能な外部記憶装置を提供することにある。

【0008】本発明の他の目的は、上位装置の側に記憶制御装置の冗長構成を意識させることなく、記憶制御装置の多重化による信頼性の向上、さらには記憶制御装置の多様な制御動作を実現することが可能な外部記憶装置を提供することにある。

【0009】本発明のさらに他の目的は、稼働を停止させることなく、冗長構成の複数の記憶制御装置におけるハードウェアおよびソフトウェア等の保守管理作業を簡便に遂行することが可能な外部記憶装置を提供することにある。

【0010】本発明のさらに他の目的は、単一のボード上に記憶制御装置およびデータバッファを搭載した構成の保守管理作業を、稼働中に実行することが可能な外部記憶装置を提供することにある。

【0011】

【課題を解決するための手段】本発明の外部記憶装置は、複数の記憶制御装置が上位装置からみて同一に見えるように記憶制御装置を上位装置に接続するインターフェイス手段と、複数の記憶制御装置間で他の記憶制御装置を監視する監視手段と、記憶制御装置間での情報の伝達が可能な通信手段と、上位装置からの要求を受領している記憶制御装置を切り替える切替手段と、1つの記憶制御装置が受領した上位装置からの入出力要求と、それに付随する処理を複数の記憶制御装置にて負荷分散する負荷分散手段とを含む構成としたものである。

【0012】また、個々の記憶制御装置に備えられ、記憶制御装置と同様の冗長構成をとることによって上位装置からの書き込みデータを一旦格納するデータバッファと、上位装置からの書き込みデータをデータバッファに

格納した時点で上位装置へ終了を報告し、上位装置との要求とは非同期にデータバッファから記憶装置に書き込むとともに、冗長構成の複数のデータバッファのすべてに書き込むか選択的に書き込むかを制御するデータ転送手段とを含む構成としたものである。

【0013】また、複数の記憶制御装置から共通にアクセス可能にされ、個々の記憶制御装置が健全か否かを識別するための第1の管理情報、ライトアフト処理およびライトスルー処理の何れを実行するかを指定する第2の管理情報、複数の記憶制御装置の何れが上位装置からの入出力要求を受け付けるかを指定する第3の管理情報、複数の記憶制御装置の各々における負荷の分担を指定する第4の管理情報、の少なくとも一つが格納される管理情報記憶手段を含む構成としたものである。

【0014】

【作用】本発明の外部記憶装置では、たとえば、一对の第1および第2の記憶制御装置を含む冗長構成であるとき、この第1および第2の記憶制御装置は上位装置と、たとえばSCSIインターフェースによりデージーチェーン接続され、同一SCSI IDでアクセスされる。たとえば上位装置からの入出力要求を固定的に第1の記憶制御装置が受領している場合、当該入出力要求の処理に伴う負荷は、複数の記憶制御装置の各々における負荷の分担を指定する第4の管理情報に基づいて他の第2の記憶制御装置に分散され、冗長構成による信頼性の向上と第1および第2の記憶制御装置の並行稼働による入出力処理の処理能力の向上が図れる。

【0015】また、たとえば上位装置からの入出力要求を固定的に第1の記憶制御装置が受領している場合、第1の記憶制御装置に障害が発生した時、第1の管理情報および監視手段により障害を検出し、切替手段によって要求を受領する記憶制御装置を第2の記憶制御装置に切り替えることにより、上位装置は障害後も同一のSCSI IDに対してI/O要求を発行すればよく、切り替えに対してなんら意識する必要はない。その後、障害となった記憶制御装置を切り放し、縮退運転に入る。部品やマイクロプログラムの交換等の保守作業終了後に第1の記憶制御装置を復旧し、元の冗長構成に復元される。

【0016】また、各記憶制御装置内にデータバッファを持ち、上位装置からの書き込みデータを第1の記憶制御装置が受領し、ライトアフト処理を実行している場合、第1の記憶制御装置に障害が発生したことを監視手段により検出し、切替手段により、第1の記憶制御装置から第2の記憶制御装置に入出力要求を受領する記憶制御装置を切り替えた時、同時に、複数のデータバッファに対する多重なデータ書き込み処理から、稼働中の記憶制御装置に備えられたデータバッファに選択的にデータを書き込む処理に切り替える。

【0017】この時、ライトアフト処理を実行するかライトスルー処理を実行するかを選択する。この選択は、

管理情報記憶手段の第2の管理情報をユーザが設定することにより可能である。すなわち、ユーザのデータ信頼性に対する要求が高い時は、ライトスルーモードに設定し、信頼性よりは性能を要求する場合は、ライトアフタモードに設定する。

【0018】第2の記憶制御装置の復旧後、選択的な書き込みか多重書き込みに切り替えることによりデータバッファに多重に書き込む操作に切り替え、冗長構成に復元できる。

【0019】上位装置からの入出力要求を第1の記憶制御装置が受領し、上位装置からの書き込み要求に関しては、第1の記憶制御装置のデータバッファと第2の記憶制御装置のデータバッファに2重書きを行い、ライトアフタ処理を行なっている場合、多重に書き込む処理から選択的に書き込む処理に切り替えて第2の記憶制御装置を切り放して縮退させ、データバッファの増設やマイクロプログラムの交換等の保守を行なった後、元の冗長構成に復旧させる。その後、記憶制御装置間での情報の伝達が可能な通信手段を用いて第2の記憶制御装置は第1の記憶制御装置にデータバッファの増設等の保守作業の完了を通知し、通知後、切替手段を用い、上位装置からの要求の受領を自装置に切り替える。

【0020】一方、通知を受けた第1の記憶制御装置は自装置を縮退させ、データバッファの増設等の保守を行ない復旧させる。復旧後、情報の伝達が可能な通信手段を用いて第1の記憶制御装置は第2の記憶制御装置にデータバッファの増設等の保守完了を通知する。これを契機に、単一のデータバッファに対する選択的なデータ書き込みから、複数のデータバッファに対する多重書き込み処理に切り替える。これにより、一対の第1の記憶制御装置および第2の記憶制御装置のデータバッファの増設やマイクロプログラムの交換等の保守管理業務を、上位装置との間における入出力処理を継続しながら可能となる。

【0021】また、本発明によれば、管理情報記憶手段に設定されている複数の前記憶制御装置の何れが前記上位装置からの入出力要求を受け付けるかを指定する第3の管理情報を参照することにより、第1の記憶制御装置および第2の記憶制御装置は、上位装置からの要求をいずれが受領するかを判断することが可能である。これにより、たとえば、第1の記憶制御装置および第2の記憶制御装置のどちらか一方のみが上位装置からの要求を受け付けることに限らず、両方の記憶制御装置にて入出力要求を受け付けて処理することも可能である。また、第3の管理情報をユーザにて随意に設定することにより、上位装置からの要求を受ける記憶制御装置をユーザから任意に指定することが可能となる。

【0022】

【実施例】以下、本発明の実施例を図面を参照しながら詳細に説明する。

【0023】図1は本発明の一実施例である外部記憶装置を含む計算機システムの一例を示す概念図である。本実施例の計算機システムは、中央処理装置である上位装置100と、ディスクドライブ制御装置200、ディスクドライブ制御装置400と、ディスク装置500とを含んでいる。ディスクドライブ制御装置200とディスクドライブ制御装置400は上位装置100とSCSIインターフェースのデージーチェーンで接続され、ディスクドライブ制御装置200、400は同一のSCSI IDが設定され、冗長構成をとっている。そして本実施例の場合、ディスクドライブ制御装置200は、上位装置100からの要求を受領し、要求に付随する処理をディスクドライブ制御装置200と冗長なディスクドライブ制御装置400とで実行し、ディスク装置500を制御する。

【0024】図2は、ディスクドライブ制御装置200および400の内部構成の一例を示すブロック図である。なお、ディスクドライブ制御装置200とディスクドライブ制御装置400の内部構成は同一であるため、ディスクドライブ制御装置200を例に説明し、ディスクドライブ制御装置400の側については対応する部位の符号の下2桁を同一にして説明は割愛する。

【0025】マイクロプロセッサユニット250（以下MPUと称す）は、ランダムアクセスメモリ（RAM、図示せず）を逐次デコードしながら実行し、ディスクドライブ制御装置200の全体を制御している。

【0026】ホストI/F制御部210は、上位装置100とのプロトコル制御を行なっている。DRV I/F制御部270は各ドライブとのプロトコル制御を行なっている。データバッファ240は、ホストI/F制御部210とDRV I/F制御部270のデータ転送時に用いられるものである。このメモリは揮発メモリでもよいし不揮発メモリでもよい。本実施例は揮発メモリでデータバッファ240を構築した場合を例に記述する。

【0027】切替機構220は、各ディスクドライブ制御装置200およびディスクドライブ制御装置400のホストI/F制御部210およびホストI/F制御部410に対し、上位装置100からのI/Oを受領するホストI/F制御部を切り替えるためのものである。この実施例では、ホストI/F制御部210が受領しているものとする。データ転送制御部230は上位装置100とデータバッファ240とのデータ転送を制御している。このデータ転送制御部230は上位装置100からのライトデータをデータバッファ240とデータバッファ440の2面に2重書きするか、データバッファ240のみの1重書きするかの両方の機能を備えている。また、MPU250からの指示により1重書きか2重書きかを切り替えることが可能である。

【0028】DRV転送制御部260はデータバッファ240とディスク装置500との間のデータ転送を制御

する。

【0029】通信機構300は、MPU250と、MPU450間での情報の伝達をするための機構である。この通信機構300はMPU250とMPU450間における双方向の伝達を可能としている。

【0030】共通管理テーブル310は、MPU250とMPU450の双方から参照/更新が可能な管理テーブルである。

【0031】本実施例では、上位装置100からの論理データを複数のディスク装置500へ分散させて格納する、アレイ構成によるドライブ格納方式を例に採って説明する。

【0032】ECC生成回路280は、上位装置100より送られてきたデータに対して冗長データを生成する機能を有し、この機能はデータの復元にも用いることができる。冗長データを付加する単位は、上位から送られてきた1論理データ単位でもよいし複数の論理データ単位に対してでもよい。本実施例は、4つの論理データに対し冗長データを付加し、この冗長データを格納するドライブを固定しないRAID5方式において記述する。

【0033】次に、図3を参照して共通管理テーブル310の構成の一例について説明する。監視情報320は各ディスクドライブ制御装置200/400が正常に動作しているかどうかをチェックするのに用いられる。監視情報A321はディスクドライブ制御装置200のMPU250が正常に動作可能と判断された時、一定間隔にて情報を設定する。また、MPU250が正常に動作不能と判断した時、異常を示す情報を設定する。なお、ディスクドライブ制御装置400のMPU450も、MPU250と同様に情報を監視情報B322に設定する。

【0034】データ転送モード情報330は、システムの縮退状態時に上位装置100からのライトデータ書き込み要求に対する終了報告契機を指示する。すなわち、この情報がデータバッファ240、またはデータバッファ440に対する書き込み完了時点で上位装置100に終了を報告するか（以下ライトアフタモードと称す）、データバッファ240からディスク装置500にまで書き込んだ時点で終了報告するか（以下ライトスルーモードと称す）を判断するための情報である。

【0035】ホストI/O受信情報340は2つのディスクドライブ制御装置200/400の内、I/Oを受信するディスクドライブ制御装置の指示情報が示されている。本実施例では、ディスクドライブ制御装置200がホストI/O受信側に設定されているものとして説明する。

【0036】負荷分散情報350は、上位装置からのI/Oに伴う処理を、2つのディスクドライブ制御装置200/400間で負荷分散するための情報である。負荷分散の方法は各ディスクドライブ制御装置にアクセス対

象とするディスク装置を分割してもよいし、上位装置100からのI/O要求の処理と、上位装置100からのI/O要求とは非同期のデータバッファからディスク装置500へライトデータを格納する処理とに分担してもよい。または処理しなければならない事柄を全て負荷分散情報の中に書き込み、2つのMPU間で競争論理とし、MPUとして空きがあるほうが処理を実行するという方法でもかまわない。

【0037】本実施例では上位装置100からのI/O要求の処理と、上位装置100からのI/O要求とは非同期にデータバッファ240/440からディスク装置500へライトデータを格納する処理とに分担する方式について説明する。よって、本実施例では、負荷分散情報350にはデータバッファ240/440に格納されたライトデータの情報が入っているものとする。

【0038】次に本実施例における計算機システムでの、上位装置100からディスク装置500に対するデータの書き込み処理および読み込み処理について説明する。

【0039】ディスクドライブ制御装置200は、通常、上位装置100からの書き込み要求時、ホストI/F制御部210により、書き込み論理データを受領し、データ転送制御部230にてデータバッファ240とデータバッファ440に2重に格納し、共通管理テーブル310の負荷分散情報350に格納情報を設定し、この時点で上位装置100に終了を報告する。MPU450は、逐次、負荷分散情報350を参照し、格納情報があれば、当該ライトデータと同一アドレスの、既にドライブに格納されているデータ（以下旧データと称す）と、当該ライトデータに対応するパリティデータをDRV I/F制御部470とDRV転送制御部460によりディスク装置500から読み出し、ECC生成回路480にてライトデータと旧データとパリティデータにて、ライトデータに対応したパリティデータ（以下新パリティデータと称す）を生成する。生成された新パリティデータとライトデータをDRV I/F制御部470とDRV転送制御部460によりディスク装置500に書き込むことにより、ライトデータをディスク装置500に格納する。この処理は上位装置100からのI/O要求とは非同期に行なわれる。また、ライトデータを格納するために行なわれる、旧データ/旧パリティデータの読み出し処理及び新パリティ生成処理、新パリティデータ格納処理はRAID5におけるライトペナルティと呼ばれている。

【0040】このように、上位装置100からのライトデータの格納要求は、複数のディスク装置500をディスクアレイ装置として機能させる場合において、非常に負荷が高い処理である。この処理を2つのディスクドライブ制御装置200、400にて役割分担し実行することは、1台のディスクドライブ制御装置だけで実行する

より、効率がよくシステムとしての性能向上につながる。特に最近の市場動向としては、安価なプロセッサを搭載し、システム全体のコストを低減させることが、高性能、高信頼性ととも、非常に大切な要素となっている。よって、ライトペナルティ処理においては、ドライブへのアクセスが多数発生することも性能劣化につながるが、それ以前にそれを制御するプロセッサのマイクロプログラムの走行時間が長いために、システムとしてプロセッサネックになることも多い。この時、本実施例のように、2台のディスクドライブ制御装置200および400にて処理を行なうことで2倍近くの性能を出すことができる。

【0041】次に上位装置100からの読み込み要求時、MPU250は、DRVI/F制御部270とDRV転送制御部260により物理ドライブ（ディスク装置500）よりデータの読み込みを開始し、上位装置100に転送する。また、この時、上位装置100からのリード要求アドレスが連続していた時、ディスクドライブ制御装置400がシーケンシャルリード処理だと判断し、上位装置100からのリード要求アドレスに続くあるデータ量を上位装置100からのI/Oとは非同期にデータバッファ240および440に読み出す処理を行っても良い。こうすることにより、次に上位装置からI/O要求があった時、対象となるデータがすでにデータバッファ240/440に格納されており、時間の掛かるディスク装置500へのアクセスを生じることなくデータを転送することができ、全体としての性能向上につながる。

【0042】以上のように、冗長構成でありながら、冗長な部分（本実施例では、ディスクドライブ制御装置400）を障害発生時の切り替え用として単にスタンバイさせておくのではなく、処理の一部を実行させることにより、信頼性だけでなく性能の向上にもつながる。

【0043】次に本実施例において、2台のディスクドライブ制御装置200/400が処理を実行しながら、障害時の自動切り替えおよび復旧を実行する動作について説明する。まず、自動的に障害を検出する監視手続きについて説明する。

【0044】MPU250、450はディスクドライブ制御装置200、400を制御しながら、一定時間が経過する度に、MPU250は監視情報321に、MPU450は監視情報322に正常であることを示す情報（以下、正常情報と称す）を設定する。但し、一定時間毎に設定していることを示すために、この情報には逐次変化する情報を設定する。たとえば、1つずつ加算されるような情報である。また、各MPU250、450が、当該ディスクドライブ制御装置200、400にて正常に動作が不可能と判断したと、たとえば、MPUからデータバッファがアクセス不可能となった時、監視情報に障害であることを示す情報（以下これを障害情報と

称す）を設定する。以下、図4のフローチャートにより前述の監視手続きの一例を説明する。

【0045】ここでは、ディスクドライブ制御装置400のMPU450が他系のディスクドライブ制御装置200の監視を行う動作を例に採り説明する。

【0046】まずMPU250はステップ600にて一定時間が経過したかを判断する。一定時間が経過していなければ、ステップ608へ進み、ディスクドライブ制御装置200が正常と判断する。

【0047】一定時間が経過していれば、ステップ601へ進み、MPU450が正常であることを示す正常情報を設定する。そしてステップ602へ進み、ディスクドライブ制御装置200の監視情報322を参照する。この情報が正常か否かを判断し、ステップ603にて正常だと判断したら、ステップ604に進む。障害であると判断したら、ステップ607に進み、ディスクドライブ制御装置200は障害であると判断する。

【0048】正常情報の時、ステップ605に進み、この正常情報に以前から変更があったかどうかをステップ605にて判断する。すなわち、MPU250がマイクロプログラムの障害等により、監視情報を設定不可能に陥っている可能性がある。このような障害を、このステップ605のチェックにて判断する。変更があれば、ステップ608へ進み、正常と判断する。変更が無かったとき、ステップ606に進み、一定時間よりも長いマージンの時間が経過しているか否かを判断する。その結果、経過していれば、ステップ607へ進み、障害と判断し、経過していなければ、ステップ608へ進み、正常と判断する。以上のような監視手続きによれば、ハードウェアの障害も、マイクロプログラムの障害も両方向同時に検出が可能である。

【0049】次に、図5のフローチャートを参照してディスクドライブ制御装置400が他系のディスクドライブ制御装置200の障害を認識して切り替わる処理の一例を説明する。

【0050】まずMPU450はステップ700にて逐次、負荷分散情報350を参照している。その結果、ステップ701にてデータバッファ240、440内に上位装置100からのライトデータが存在しなければ、ステップ704に進む。存在すれば、ステップ702に進み、データバッファ440のライトデータに対応するパリティを生成するため、当該ライトデータに対応する旧データと旧パリティデータをディスク装置500から読み出し、ECC生成回路480にて新パリティデータを生成する。その後、ステップ703に進み、ライトデータと新パリティデータをDRV転送制御部460および、DRVI/F制御部470によりディスク装置500に格納する。次にステップ704にて、図4のステップ600以降の監視手続きによりディスクドライブ制御装置200の障害をチェックする。その結果、正常なら

ば、ステップ700に進み、処理を続ける。切り替えが必要と判断したら、ステップ710に進み、切り替え手続きを用いて、上位装置100からのI/Oの受信をディスクドライブ制御装置200からディスクドライブ制御装置400に切り替える。そして、ディスクドライブ制御装置200が行っていた上位装置100からのI/O処理をステップ720にてディスクドライブ制御装置400が代替して行なう。

【0051】次に、図6のフローチャートにて切り替え手続きの一例を説明する。

【0052】まずステップ711にて、データ転送制御部430に対し、上位装置100からのライトデータ受領時、当該データをデータバッファ440へ1重に書き込むことを指示する。すなわち、ディスクドライブ制御装置200に障害が発生したため、ディスクドライブ制御装置200を縮退させて切り放し、障害発生した部位を交換し、復旧するまでの間、データバッファはディスクドライブ制御装置400にしか存在しないため、正常な冗長構成時のような2重書きはできない。

【0053】そして、ステップ712にて、切替機構420にて上位装置100からのI/O要求をホストI/F制御部210から、ホストI/F制御部410に切り替えるよう指示をする。これにより、ホストI/F制御部210は上位装置100からの要求を受け付けなくなり、また、ホストI/F制御部410は上位装置100からの要求を受け付けるようになり、実質的にはディスクドライブ制御装置が切り替わることになるが、本実施例ではSCSIDが同一なため、上位装置100はI/Oを切り替える以前と同様のSCSIDに発行すればよく、受領側のディスクドライブ制御装置が切り替わったことを知る必要が全くない。

【0054】次に図7のフローチャートを用いて、切り替わった後、ディスクドライブ制御装置400にてI/Oを実行する手順の一例を説明する。

【0055】ステップ721にて上位装置100よりI/O処理を受信すると、ステップ722に進み、リード要求かライト要求かを判断する。リード要求の時、ステップ729に進み、当該リード要求に対応するディスク装置500よりデータバッファ440に対象データを読み込む。ステップ730に進み、データバッファ440から上位装置100へデータを転送し、ステップ728にて上位装置100に対し、終了を報告する。

【0056】ライト要求の時、ステップ723に進み、データバッファ440にライトデータを格納する。さらに、ステップ724に進み、データ転送モード情報330を参照し、ステップ725でライトスルーモードか否かを判定する。その結果、ライトアフトモードの時、すなわちデータバッファ440に格納した時点で上位装置100に対して終了を報告するモードの時は、ステップ728に進み、終了を報告し、その後、非同期にデータ

バッファ440からディスク装置500に格納する。ライトスルーモードの時は、ステップ726に進み、ライトデータに対するパリティデータを作成し、ステップ727にてライトデータと新パリティデータをディスク装置500に格納し、ステップ728にて終了を報告する。さらに、この後、図5のフローチャートにおけるステップ700からステップ703を実行し、切り替え以前の処理も実行する。

【0057】このように、本実施例によれば、上位装置100からの指示なしに、上位装置100になんの意識もさせずに、ディスクドライブ制御装置200/400にて、自動的に相互間の切り替え動作および処理の続行が可能である。

【0058】次に、ディスクドライブ制御装置200が復旧し、元の冗長構成に戻る時の方法の一例を説明する。

【0059】まず、図8に示されるフローチャートにて、ディスクドライブ制御装置200の側の復旧動作の一例について説明する。ステップ810で、通信機構300にてディスクドライブ制御装置400に対して復旧が完了したことを通知する。その後、ディスクドライブ制御装置200が冗長なディスクドライブ制御装置となり、以前と立場が入れ替わる。ステップ811にて、以前、ディスクドライブ制御装置400が行っていた非同期のデステージ処理（図5のステップ700～705）を行なう。

【0060】さらに、図9に示されるフローチャートにて、通知を受けた側のディスクドライブ制御装置400の動作の一例を説明する。

【0061】ステップ820にて通信機構300にてディスクドライブ制御装置200の復旧完了を認識すると、ステップ821でデータ転送制御部430にデータバッファ240と440への2重書きを指示し、ステップ821にて上位装置100からのI/O処理のみを実行する。このように、上位装置100からのI/O要求を受けながら、元の冗長な構成に復旧することが可能であり、さらに2つのディスクドライブ制御装置200/400で処理を負荷分散することにより、性能の向上も図れる。

【0062】次に、ディスクドライブ制御装置200/400の稼働中におけるデータバッファの増設方法の一例について図10のフローチャートを参照しながら説明する。なお、上位装置100からのI/Oを受信しているディスクドライブ制御装置は、ディスクドライブ制御装置200とする。

【0063】データバッファの増設要求があった時、ステップ911にて当該ディスクドライブ制御装置はI/O受信側かを判断する。まずディスクドライブ制御装置200の処理内容について説明する。ディスクドライブ制御装置200はI/O受信側なので、ステップ912

に進み、ディスクドライブ制御装置400がまず縮退し、切り放すことを認識する。そこで、データ転送制御部230にデータバッファ240へライトデータを1重書きとするよう指示をする。その後、ステップ913にて上位装置100からのI/O処理を実行し、ステップ914にて、図5のステップ700～ステップ703を実行する。すなわちディスクドライブ制御装置400にて実行していた分を代替する。ステップ913とステップ914を繰り返しながらディスクドライブ制御装置400の復旧完了を待つ。

【0064】次にディスクドライブ制御装置400もステップ911にて当該ディスクドライブ制御装置はI/O受信側かを判断する。その結果I/O受信側ではないので、ステップ915に進み、当該ディスクドライブ制御装置400は切り放しを行ない、ステップ916にてデータバッファ440の増設を行なう。増設完了後、ステップ917にて復旧したことを通信機構300を介してディスクドライブ制御装置200に通知する。

【0065】今度はディスクドライブ制御装置200が増設を行なう必要があるため、ディスクドライブ制御装置400はI/Oの受領を代替するため、ステップ919で切替機構420を用いてI/O受信するホストI/F制御部を自系に切り替える。その後、ステップ920にて上位装置100からのI/O処理を実行し、ステップ921にて図5のステップ700～ステップ703を実行し、ディスクドライブ制御装置200の復旧完了を待つ。

【0066】ステップ918で復旧を通信機構300を介して認識したディスクドライブ制御装置200は、ステップ922にて切り放し、ステップ923にてデータバッファ240の増設を行なう。増設完了後、ステップ924にて復旧を通信機構300にてディスクドライブ制御装置400に通知する。通知後、当該ディスクドライブ制御装置200はホストI/O受信側ではないのでステップ925にて図5のステップ700～ステップ705を実行する側に回る。

【0067】ディスクドライブ制御装置400は、ステップ926にて他系の復旧を通信機構300にて認識すると、ステップ927にてデータ転送制御部230にデータバッファ240/440へライトデータを2重に書くよう指示をする。ステップ928にて上位装置100からのI/O処理を実行する。

【0068】このように上位装置100からのI/Oを実行しながらも各系のデータバッファ240/440の増設が可能となる。すなわち本実施例によれば、従来ではデータバッファの増設はシステムを停止してからでないと実現できなかったのに対し、オンライン中に増設が可能となる。特に、低コストにて実現されている1つのボード上にディスクドライブ制御装置が構築されているときは、ボード毎の交換が必要なため、稼働中の増設は

不可能であった。本実施例では、冗長構成のディスクドライブ制御装置200/400において、1台ずつを縮退/復旧させながら、データバッファの増設が可能である。

【0069】また、本実施例によれば、図10のステップ916および、ステップ923の処理をマイクロプログラム交換作業に置き換えることにより、稼働中のマイクロプログラムの交換が可能であり、24時間運転の要求が著しい近年のコンピュータシステムにおける保守管理作業に特に有効である。

【0070】また、片系障害時の縮退中、上位装置100からのライト要求をデータバッファまでに書き込んで終了を報告するか、ディスク装置500にまで書き込んで終了を報告するかはユーザが指示可能である。すなわち、このデータ転送モード情報330の書き換えは、ユーザのプログラムで自動的に行なってもよい。すなわち、データバッファが1面構成になった時、データファックに格納した時点で終了を報告すれば、応答性にはすぐれているが、この時点でディスクドライブ制御装置に障害が発生すると、データ保証ができなくなる。一方、ディスク装置500にまで格納するのでは、ライトペナルティ処理が発生してしまうため、応答性はかなり劣化してしまうが、上位装置100に対しては、確実な応答が報告でき、信頼性は高い。本実施例の外部記憶装置の場合、ユーザが扱うファイルへの信頼度の要求レベルに応じて、ユーザの指示により、信頼度を優先するか、応答速度を優先するかを随意に選択でき、柔軟なファイルシステムを構築することが可能となる。

【0071】さらに本発明では、複数のディスクドライブ制御装置は冗長な構成だけではなく、複数の上位装置または、複数のバスより、同時にアクセスが可能なシステムも提供できる。このシステム構成例を図11および図12に示す。

【0072】図11は、これまでに説明した実施例の図1と同じ構成だが、上位装置100とのI/FがSCSIの時、図1の構成では記憶制御装置0と記憶制御装置1は同じSCSI IDで接続されていたのに対して、図11の構成では記憶制御装置0(400A)と記憶制御装置1(200A)は異なるSCSI IDで接続されている点が異なっている。この図11の構成の場合、どちらも、上位装置100からI/O要求を受領して処理する。また、図12は、複数の記憶制御装置0(400B)および記憶制御装置1(200B)が、同一の上位装置100に対してマルチバスにより接続されたシステム構成の一例を示すブロック図である。この図12の構成でも、記憶制御装置0(400B)と記憶制御装置1(200B)は、いずれも上位装置100からのI/O要求を実行可能である。いずれがI/O要求を実行するかの指定は共通管理テーブル310の、ホストI/O受信情報340を書き換えることにより実現される。すな

わち、各記憶制御装置は、まずホスト I/O 受信情報 340 を参照し、当該記憶制御装置が上位装置からの I/O を受信するか否かを決定する。このように、本発明では様々なユーザの接続方法に対応することができ、柔軟なシステムが構築できる。

【0073】以上説明したように、本実施例によれば、冗長構成の複数のディスクドライブ制御装置 200/400 が負荷分散しながら上位装置 100 からの要求を実行することにより、信頼性の向上だけでなく性能の向上も同時に実現することが可能なファイルシステムを提供できる。また、すべてのディスクドライブ制御装置 200/400 が負荷分散しながら上位装置 100 からの I/O 要求を実行しながらでも、障害発生時に上位装置 100 からなんら指示を仰ぐことなく自動的に切り替わって稼働を継続し、さらに復旧することが可能となる。これにより、上位装置 100 からの I/O 要求を実行しながらデータバッファの増設やマイクロプログラムの交換が可能となり、無停止保守が実現できる。また、冗長構成だけでなく、すべてのディスクドライブ制御装置が同時に上位装置 100 からの要求を受信する構成にすることも可能であり、ユーザの要求する多様なファイルシステムに柔軟に対応することができる。

【0074】

【発明の効果】本発明の外部記憶装置によれば、冗長構成の複数の記憶制御装置に負荷を分散させることにより、信頼性および性能を向上させることができる、という効果が得られる。

【0075】また、上位装置の側に記憶制御装置の冗長構成を意識させることなく、記憶制御装置の多重化による信頼性の向上、さらには記憶制御装置の多様な制御動作を実現することができる、という効果が得られる。

【0076】また、稼働を停止させることなく、冗長構成の複数の記憶制御装置におけるハードウェアおよびソフトウェア等の保守管理作業を簡便に遂行することができる、という効果が得られる。

【0077】また、単一のボード上に記憶制御装置およびデータバッファを搭載した構成の保守管理作業を、稼働中に実行することができる、という効果が得られる。

【図面の簡単な説明】

【図 1】本発明の一実施例である外部記憶装置を含む計算機システムの一例を示す概念図である。

【図 2】本発明の一実施例である外部記憶装置を構成するディスクドライブ制御装置の内部構成の一例を示すブロック図である。

【図 3】本発明の一実施例である外部記憶装置において用いられる共通管理テーブルの構成の一例を示す概念図である。

【図 4】本発明の一実施例である外部記憶装置の作用の一例を示すフローチャートである。

【図 5】本発明の一実施例である外部記憶装置の作用の一例を示すフローチャートである。

【図 6】本発明の一実施例である外部記憶装置の作用の一例を示すフローチャートである。

【図 7】本発明の一実施例である外部記憶装置の作用の一例を示すフローチャートである。

【図 8】本発明の一実施例である外部記憶装置の作用の一例を示すフローチャートである。

【図 9】本発明の一実施例である外部記憶装置の作用の一例を示すフローチャートである。

【図 10】本発明の一実施例である外部記憶装置の作用の一例を示すフローチャートである。

【図 11】本発明の一実施例である外部記憶装置における上位装置との接続形態の変形例を示す概念図である。

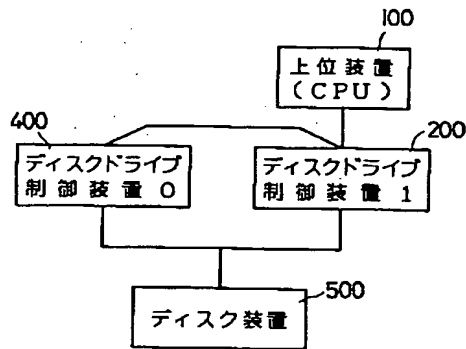
【図 12】本発明の一実施例である外部記憶装置における上位装置との接続形態の変形例を示す概念図である。

【符号の説明】

100…上位装置、200…ディスクドライブ制御装置、210…ホスト I/F 制御部、220…切替機構、230…データ転送制御部、240…データバッファ、250…マイクロプロセッサユニット、260…DRV 転送制御部、270…DRV I/F 制御部、280…ECC 生成回路、300…通信機構、310…共通管理テーブル、320…監視情報、321…監視情報、322…監視情報、330…データ転送モード情報、340…ホスト I/O 受信情報、350…負荷分散情報、400…ディスクドライブ制御装置、410…ホスト I/F 制御部、420…切替機構、430…データ転送制御部、440…データバッファ、450…マイクロプロセッサユニット、460…DRV 転送制御部、470…DRV I/F 制御部、480…ECC 生成回路、500…ディスク装置。

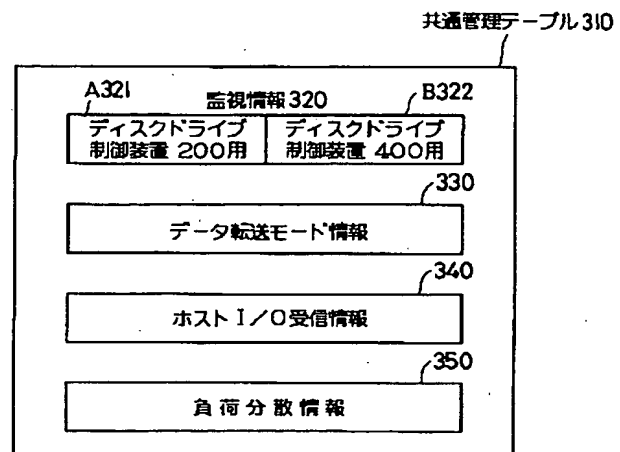
【図 1】

図 1



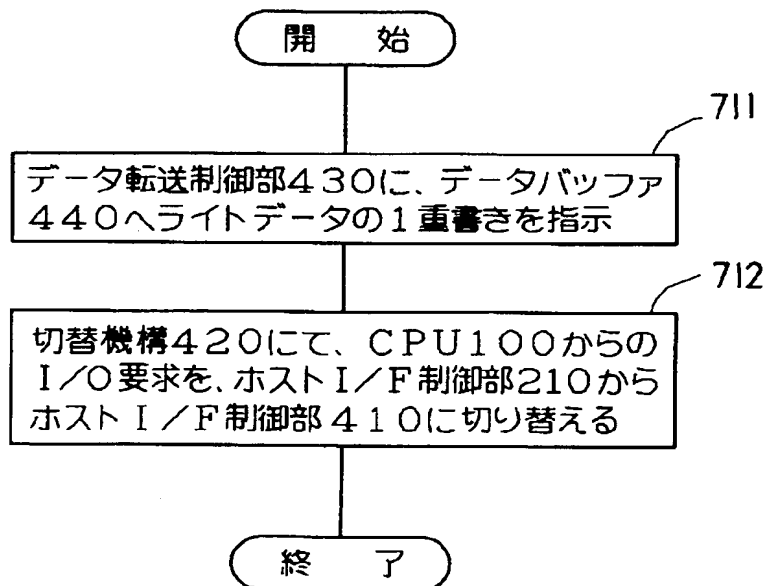
【図 3】

図 3



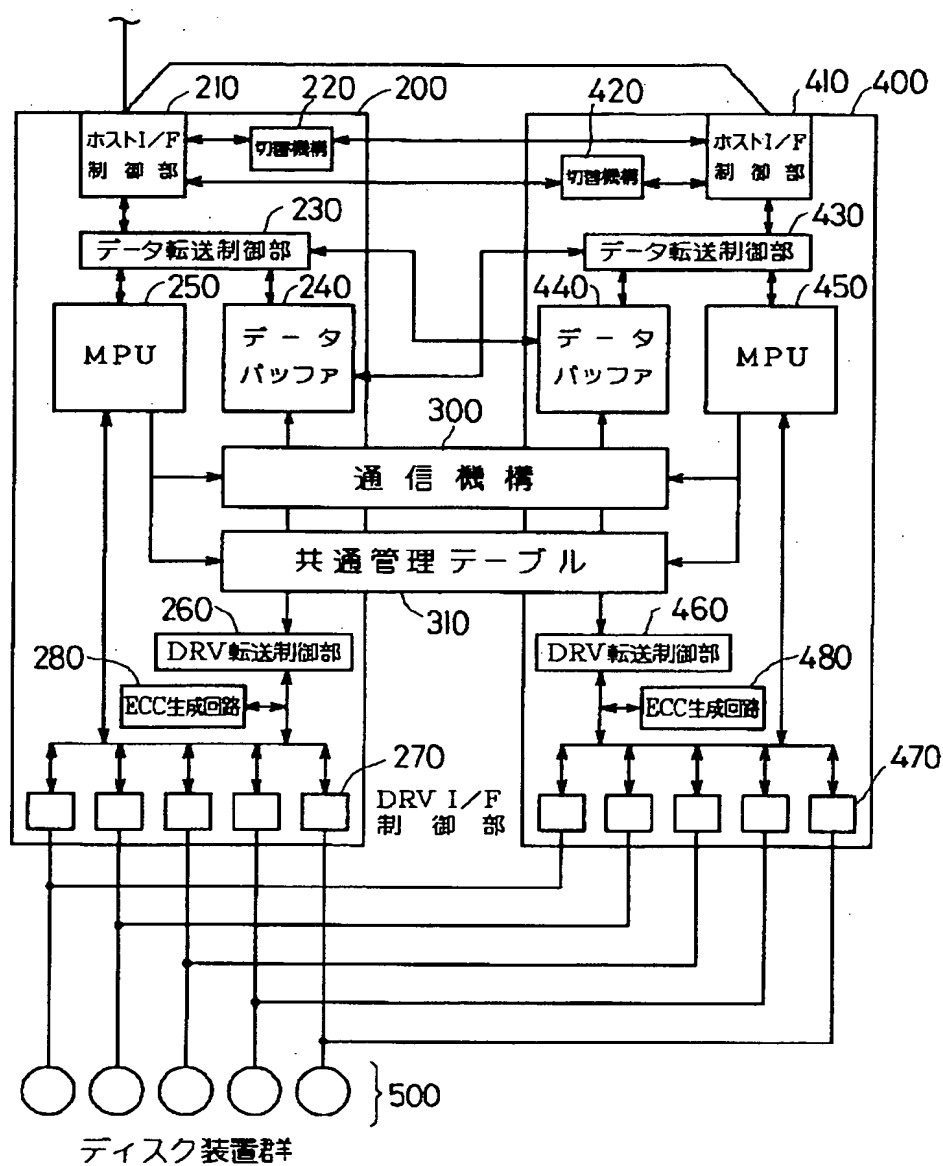
【図 6】

図 6



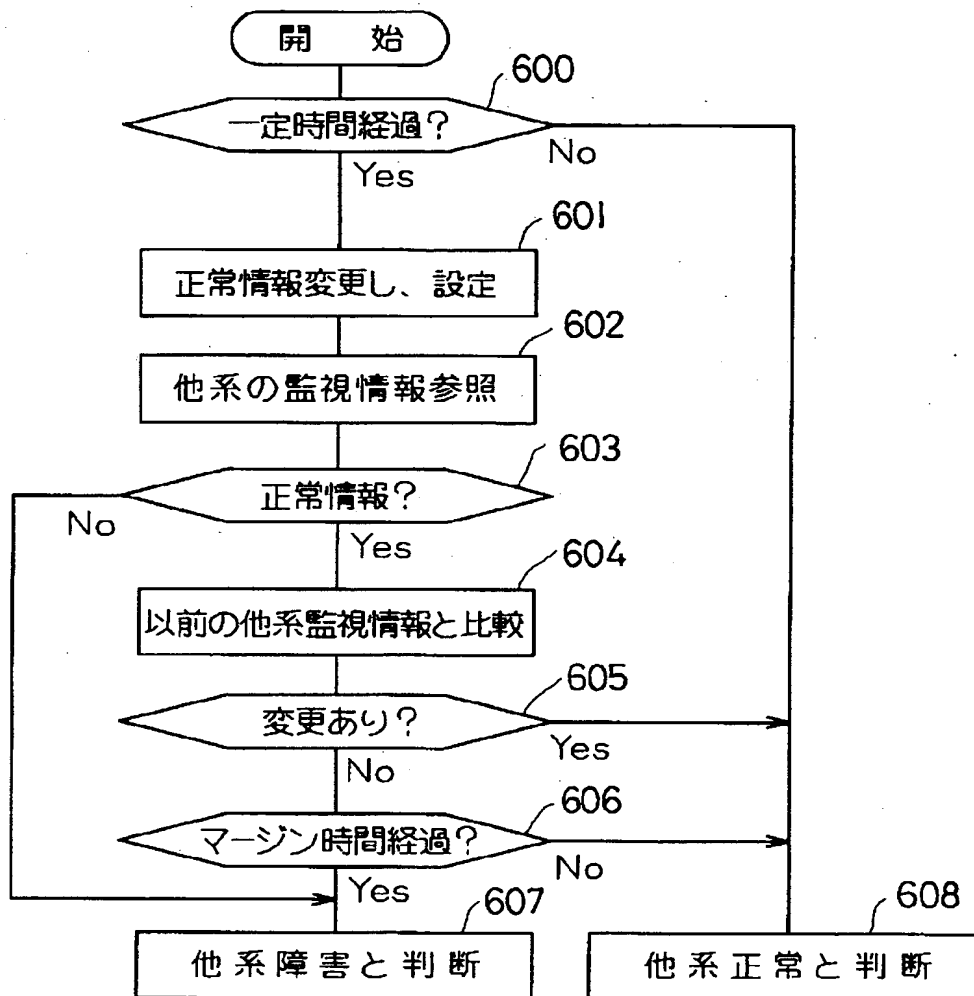
【図2】

図2



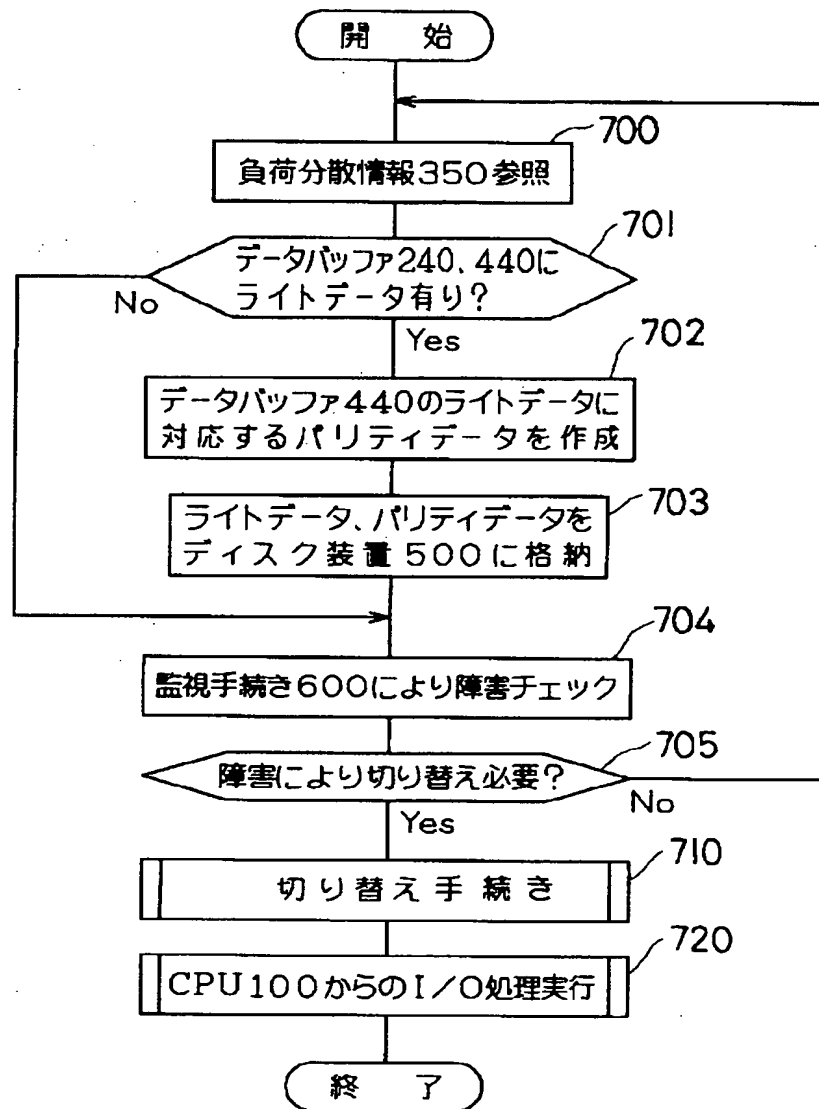
【図 4】

図 4



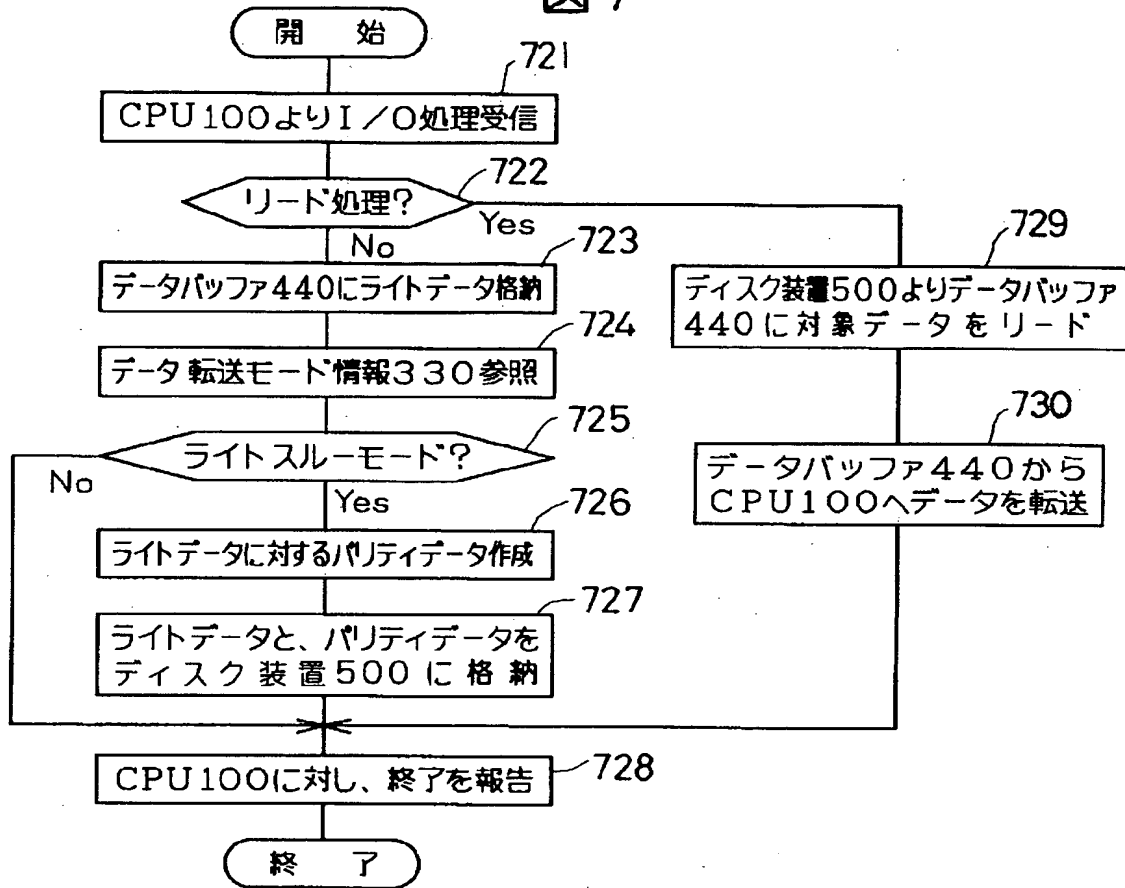
【図5】

図5



【図7】

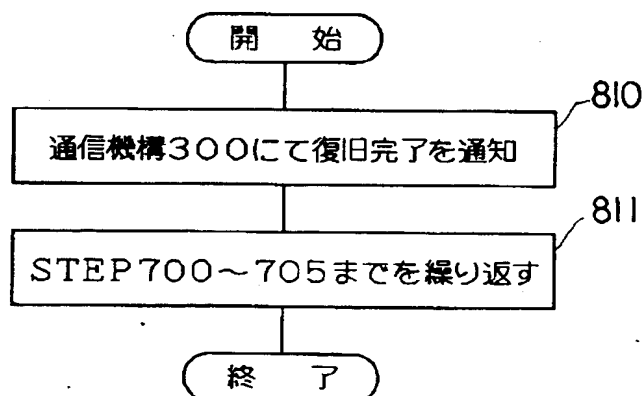
図7



【図8】

図8

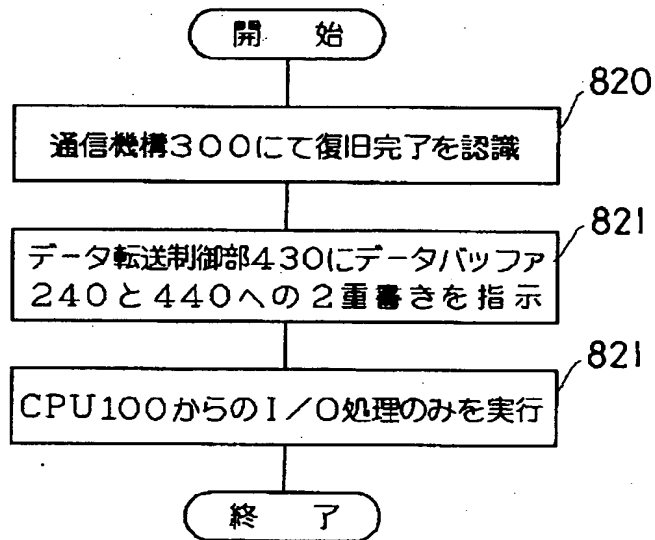
復旧方式（ディスクドライブ制御装置200）



【図 9】

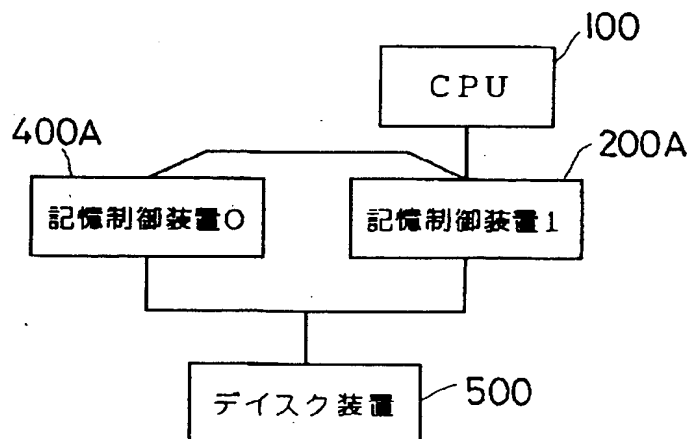
図 9

復旧方式（ディスクドライブ制御装置400）



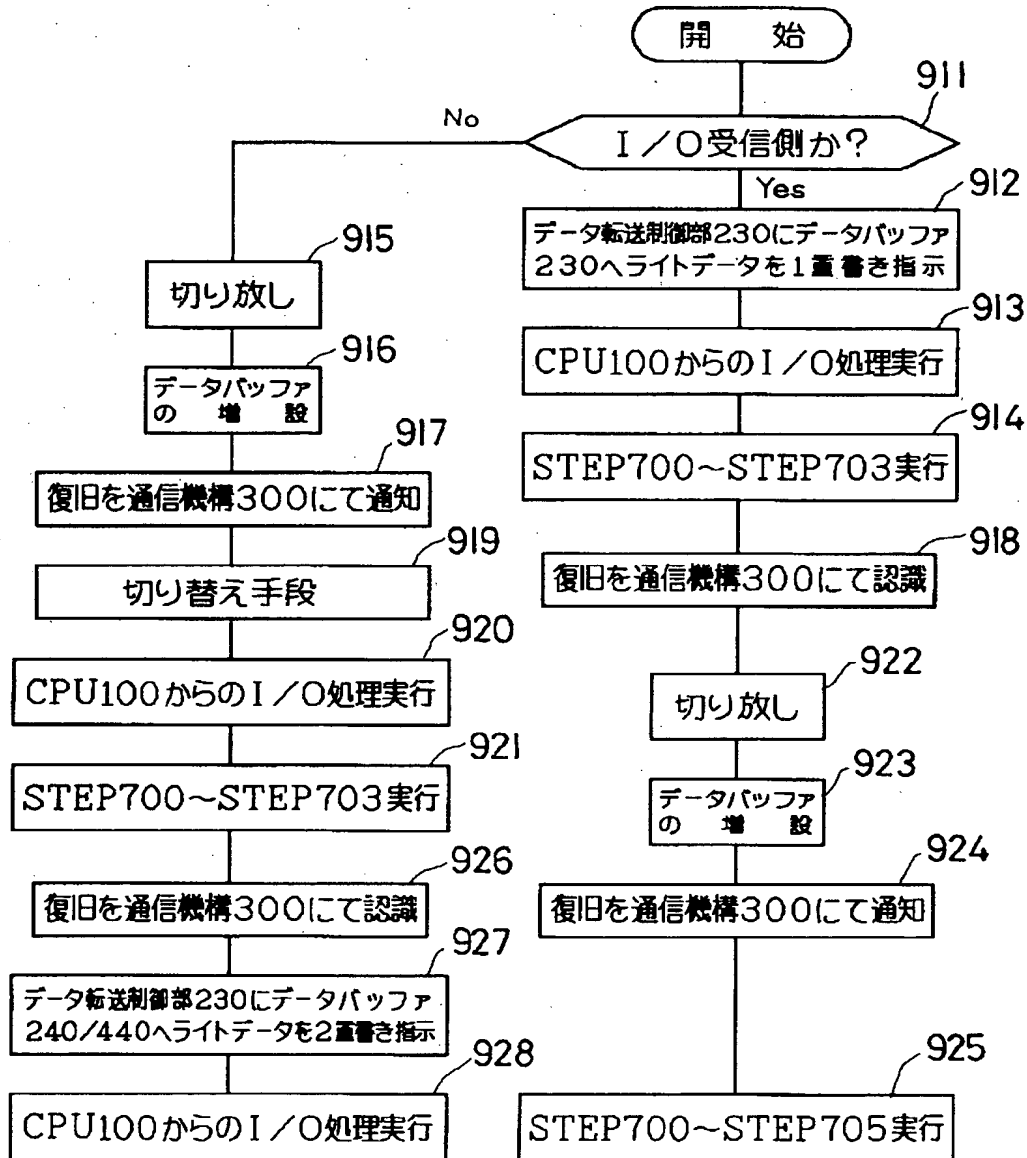
【図 11】

図 11



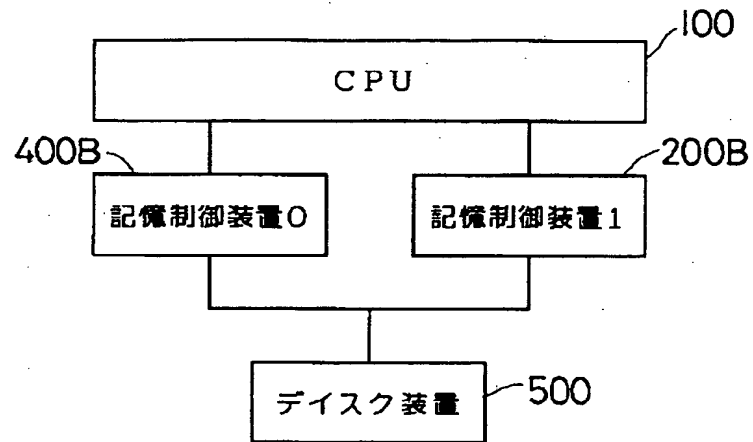
【図10】

図 10



【図 12】

図 12



【公報種別】特許法第 17 条の 2 の規定による補正の掲載

【部門区分】第 6 部門第 3 区分

【発行日】平成 14 年 8 月 30 日 (2002. 8. 30)

【公開番号】特開平 8-335144

【公開日】平成 8 年 12 月 17 日 (1996. 12. 17)

【年通号数】公開特許公報 8-3352

【出願番号】特願平 7-139781

【国際特許分類第 7 版】

G06F	3/06	304
	11/20	310
	12/16	310
	13/14	310

【F I】

G06F	3/06	304 B
	11/20	310 B
	12/16	310 Q
	13/14	310 F

【手続補正書】

【提出日】平成 14 年 6 月 6 日 (2002. 6. 6)

【手続補正 1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正内容】

【特許請求の範囲】

【請求項 1】 上位装置との間で授受されるデータが格納される記憶装置と、前記記憶装置と前記上位装置との間に介在し、前記上位装置と前記記憶装置との間における前記データの授受を制御する複数の記憶制御装置とを含む外部記憶装置であって、

複数の前記記憶制御装置が前記上位装置からみて等価に見えるように当該記憶制御装置を前記上位装置に接続するインターフェイス手段と、

個々の前記記憶制御装置に設けられ、他の前記記憶制御装置における障害または切替指令の有無を監視する監視手段と、

個々の前記記憶制御装置に設けられ、いずれの前記記憶制御装置が前記上位装置との間における前記データの授受の制御を行うかを切り替える切替手段と、

前記記憶制御装置の相互間における情報の伝達を行う情報伝達手段と、

前記上位装置からの入出力要求に起因する負荷を複数の前記記憶制御装置間にて分担させる負荷分散手段と、を備えたことを特徴とする外部記憶装置。

【請求項 2】 請求項 1 記載の外部記憶装置において、複数の前記記憶制御装置の各々に設けられ、前記上位装置との間で授受される前記データを一時的に格納するデータバッファと、

前記上位装置からの書き込み要求時、複数の前記データバッファの各々に対して書き込み要求データを選択的または多重に書き込むとともに、前記書き込み要求データの前記データバッファに対する書き込み完了時点で前記上位装置に対して書き込み完了を報告し、前記上位装置からの入出力要求とは非同期に前記データバッファから前記記憶装置へ前記書き込み要求データを反映させるライトアプタ処理、および前記書き込み要求データの前記記憶装置に対する書き込み完了時点で前記上位装置に対して書き込み完了を報告するライトスルー処理を選択的に実行可能なデータ転送制御手段と、を備えたことを特徴とする外部記憶装置。

【請求項 3】 請求項 1 または 2 記載の外部記憶装置において、

複数の前記記憶制御装置から共通にアクセス可能にされ、個々の前記記憶制御装置が健全か否かを識別するための第 1 の管理情報、前記ライトアプタ処理および前記ライトスルー処理の何れを実行するかを指定する第 2 の管理情報、複数の前記記憶制御装置の何れが前記上位装置からの入出力要求を受け付けるかを指定する第 3 の管理情報、複数の前記記憶制御装置の各々における前記負荷の分担を指定する第 4 の管理情報の少なくとも一つが格納される管理情報記憶手段と、

障害の発生または外部からの切替指令を契機として、前記障害が発生したか、または外部から指令された前記記憶制御装置を切り離すとともに、残りの前記記憶制御装置によって前記上位装置との間における前記データの授受を継続する縮退運転を行う操作、および切り離されていた前記記憶制御装置を冗長構成に復帰させる操作を行う制御論理と、

を備えたことを特徴とする外部記憶装置。

【請求項4】 請求項2記載の外部記憶装置において、前記データ転送制御手段は、個々の前記記憶制御装置の各々に設けられた前記データバッファの各々に対する前記書き込み要求データの選択的な書き込み操作の停止および再開を行う制御論理を備えたことを特徴とする外部記憶装置。

【請求項5】 請求項4記載の外部記憶装置において、複数の前記記憶制御装置の中の少なくとも一つを選択的に停止させて縮退運転を行うとともに、停止された前記記憶制御装置に対応するデータバッファの保守または前記記憶制御装置を制御するマイクロプログラムの保守を実行することを特徴とする外部記憶装置。

【請求項6】 上位装置との間で授受されるデータが格納される記憶装置と、
前記記憶装置と前記上位装置との間に介在し、前記上位

装置と前記記憶装置との間における前記データの授受を制御する第1の記憶制御装置と、

前記記憶装置と前記上位装置との間に介在し、前記上位装置と前記記憶装置との間における前記データの授受を制御する第2の記憶制御装置と、

前記記憶制御装置から共通にアクセス可能にされ、個々の前記記憶制御装置の管理情報を格納する管理情報記憶手段とを含む外部記憶装置であって、

通常は、前記第1の記憶制御装置が、前記上位装置と前記記憶装置との間で処理されるべき情報であって入出力要求を含むものを処理し、かつ、

前記上位装置と前記記憶装置との間で処理されるべき入出力要求に含まれる情報であって、前記記憶制御装置の各々における負荷の分担を指定する管理情報を、前記管理情報記憶手段に記憶することを特徴とする外部記憶装置。